SOC WORKFLOW DEPLOYMENT STRATEGY

Chapter 1: Strategy Overview and Benefits

1.1 Problem Statement

Modern e-commerce platforms face critical PII leakage vulnerabilities across multiple attack vectors:unmonitored API integrations, network ingress layers, plain-text database storage, and internal application rendering. Traditional single-point security solutions create system-wide vulnerabilities when detection mechanisms fail.

1.2 Proposed Solution Architecture

This report presents a three-layer defense system with intelligent failure routing that maintains security posture even during primary detection system failures. The architecture implements circuit breaker patterns to route traffic through quarantine servers when main PII detection engines become unavailable.

1.3 Core Benefits

Security Benefits:

- Zero single-point-of-failure in PII detection pipeline
- Graceful degradation maintains protection during system failures
- Conservative fallback ensures compliance under adverse conditions

Operational Benefits:

- Best efficient service availability during security system maintenance
- Sub-10ms latency impact on normal operations
- Automated incident response with manual override capabilities

Chapter 2: Used Concepts

2.1 Circuit Breaker Pattern

Implements Martin Fowler's circuit breaker pattern to handle PII detection service failures. Three statesmanage traffic routing:

- Closed: Normal operation, requests pass through primary detection
- Open: Detection failure detected, traffic routed to quarantine server
- Half-Open: Testing recovery, limited traffic routed to primary system

2.2 Multi-Layer Defense

Applies cybersecurity defense-in-depth principles across three architectural layers:

- API Gateway Layer: Primary real-time detection and filtering
- Log Processing Layer: Secondary analysis via Kubernetes DaemonSet
- Database Layer: Tertiary protection through transparent proxy encryption

2.3 Quarantine Processing
Borrowed from malware detection systems, implements isolated processing environment for suspicious traffic:

- Conservative detection algorithms with higher false-positive tolerance
- Aggressive redaction policies for uncertain data patterns
- Manual review queues for business-critical operations

2.4 Service Mesh Principles
Utilizes service mesh concepts for traffic management:

- Sidecar pattern for log interception
- Transparent proxy for database operations
- Policy-driven routing based on detection engine health

2.5 Stream Processing Architecture
Employs stream processing patterns for real-time data analysis:

- Event-driven processing with Kafka streams
- Stateful stream aggregation for pattern recognition
- Backpressure handling for high-throughput scenarios

Chapter 3: Architecture Explanation with Implementation

3.1 System Architecture Overview
The architecture implements three primary defense layers with intelligent routing capabilities. Each layer operates independently while contributing to overall system security posture.

3.2 Primary Defense: API Gateway Layer
Implementation: High-performance middleware deployed within existing API Gateway infrastructure using Go/Rust for optimal concurrency.

Components:

- PII Detection Engine: Hybrid regex + ML processing
- Pattern Cache: Redis-backed specific TTL for frequent patterns
- Circuit Breaker: Health monitoring with configurable failure thresholds
- Traffic Router: Intelligent routing based on detection engine status

Processing Flow:
HTTP Request → Load Balancer → API Gateway → PII Detector → Circuit Breaker → Traffic Router → Service/Quarantine

3.3 Secondary Defense: Log Sanitization Layer
Implementation: Kubernetes DaemonSet deployment across all cluster nodes for distributed log processing.
Components:

- Log Interceptor: Captures stdout/stderr and file-based logs
- Stream Processor: Real-time PII detection with a practical throughput

3.4 Tertiary Defense: Database Protection Layer

Implementation: Transparent database proxy with field-level encryption capabilities.

Components:

- Query Rewriter: Automatic encryption injection for detected PII columns
- Encryption Engine: AES-256 encryption with Hardware Security Module integration
- Key Management: Centralized key rotation and access control

3.5 Failure Routing Mechanism

Quarantine Server Architecture:

- Conservative Detection: Rule-based processing with minimal ML dependency
- Aggressive Redaction: Higher threshold for data masking decisions
- Safe-Mode Services: Feature-limited service replicas for fallback operations

NOTE: You can see deploy.png for further details