[COMPANY LOGO]

Application Intelligence Report

Comprehensive Analysis and Migration Assessment

Repository: https://github.com/end-of-game/openshift-voting-app

Analysis Date: July 17, 2025

*Generated by Application Intelligence Platform*

Table of Contents

Executive Summary........................................... 3

Application Overview........................................ 4

Component Analysis.......................................... 5

Architecture Assessment..................................... 8

Security Analysis........................................... 10

Git History Analysis........................................ 12

Recommendations............................................. 13

Appendices.................................................. 15

Executive Summary

|  |  |
| --- | --- |
| **Metric** | **Value** |
| Total Components | 8 |
| Programming Languages | java, nodejs, python |
| Containerization Status | 5 containerized |
| Data Sources | 2 |
| Security Findings | 0 |
| Git Commits | 1 |
| Architecture Style | microservices |

Application Overview

This report presents a comprehensive analysis of the application repository. The analysis identified 8 components using 3 different programming languages. The application demonstrates a microservices architecture pattern.

Key Findings

• 📦 8 application components identified

• 🔧 3 programming languages detected: java, nodejs, python

• 🐳 5 components are containerized

• 💾 2 data sources identified

• 🔒 0 security findings require attention

Detailed Analysis

Component Analysis

The analysis identified 8 components across the application:

|  |  |  |  |
| --- | --- | --- | --- |
| **Component** | **Language** | **Type** | **Packaging** |
| postgresql-ephemeral | unknown | Unknown | docker |
| redis-ephemeral | unknown | Unknown | docker |
| result | nodejs | Unknown | docker |
| vote-s2i | python | Unknown | wheel |
| result-s2i | nodejs | Unknown | npm |
| vote | python | Unknown | docker |
| worker-s2i | java | Unknown | jar |
| worker | java | Unknown | docker |

Component: postgresql-ephemeral

• Language: unknown

• Runtime: unknown

• Build Tool: unknown

• Packaging: docker

Component: redis-ephemeral

• Language: unknown

• Runtime: unknown

• Build Tool: unknown

• Packaging: docker

Component: result

• Language: nodejs

• Runtime: nodejs

• Build Tool: unknown

• Packaging: docker

• Exposed Ports: 8080

• Base Images: node:10-slim

Component: vote-s2i

• Language: python

• Runtime: python

• Build Tool: unknown

• Packaging: wheel

Component: result-s2i

• Language: nodejs

• Runtime: nodejs

• Build Tool: unknown

• Packaging: npm

Component: vote

• Language: python

• Runtime: python

• Build Tool: unknown

• Packaging: docker

• Exposed Ports: 8080

• Base Images: python:3.9-slim

Component: worker-s2i

• Language: java

• Runtime: java

• Build Tool: unknown

• Packaging: jar

Component: worker

• Language: java

• Runtime: java

• Build Tool: unknown

• Packaging: docker

• Base Images: maven:3.5-jdk-8-alpine, openjdk:8-jre

**Notes:**

• Multiple base images detected: maven:3.5-jdk-8-alpine, openjdk:8-jre. This may indicate multi-stage builds or alternative build strategies.

Architecture Analysis

Architecture Style: microservices (Confidence: ConfidenceLevel.HIGH)

Reasoning: Multiple components with independent deployment characteristics

**Evidence:**

• Found 8 components

• Multiple deployable components detected

• 5 containerized components

• Multiple deployment configurations

Security Analysis

Security analysis identified 25 findings with 3 base image risks.

**Key Security Findings:**

• Unknown: Pattern detected: eval\( (Severity: CRITICAL)

• Unknown: Pattern detected: eval\( (Severity: CRITICAL)

• Unknown: Pattern detected: eval\( (Severity: CRITICAL)

• Unknown: Pattern detected: eval\( (Severity: CRITICAL)

• Unknown: Pattern detected: eval\( (Severity: CRITICAL)

Git History Analysis

• Total Commits: 1

• Active Contributors: 0

• Recent Activity: inactive

• Code Stability: high

Recommendations

🔴 High Priority Recommendations

• 🔒 Security: 25 critical/high severity vulnerabilities found. Prioritize security remediation.

🟢 Low Priority Recommendations

• 🔍 Component Analysis: 2 components have unknown languages. Review build configurations and source code structure.

• 📊 Development Activity: Low recent activity detected. Consider reviewing development processes and team capacity.

• 🐳 Base Images: 3 base images have known risks. Update to more recent versions.

Appendices

Appendix A: Technical Details

This analysis was generated using the Application Intelligence Platform, which performs comprehensive analysis of application repositories including code structure, infrastructure configuration, and security assessment.

Appendix B: Analysis Methodology

• Component Discovery: Automated scanning of source code and configuration files

• Language Detection: Analysis of file extensions, build configurations, and base images

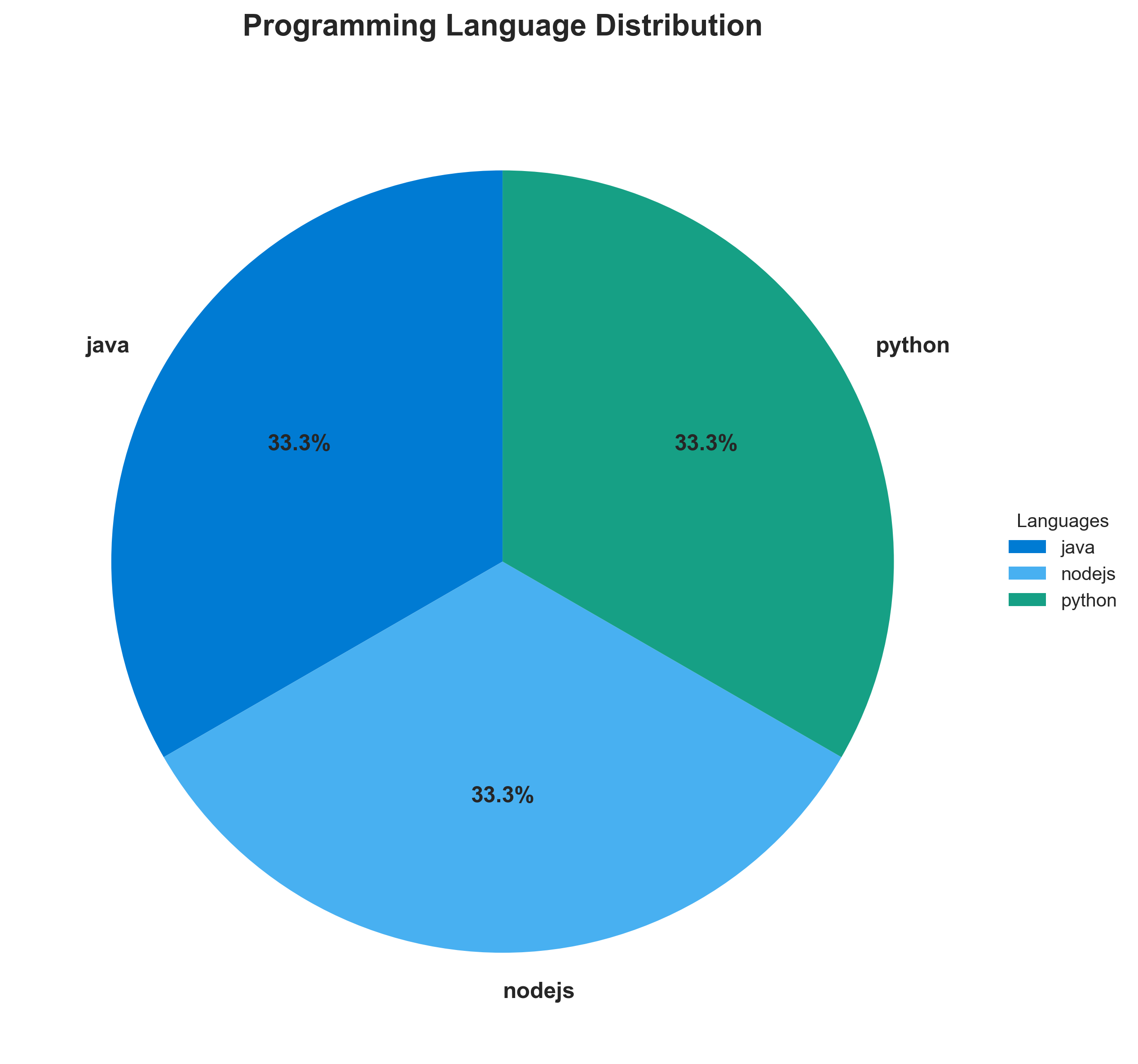
• Architecture Assessment: Evaluation of deployment patterns and component relationships

• Security Analysis: Scanning for common vulnerabilities and configuration issues

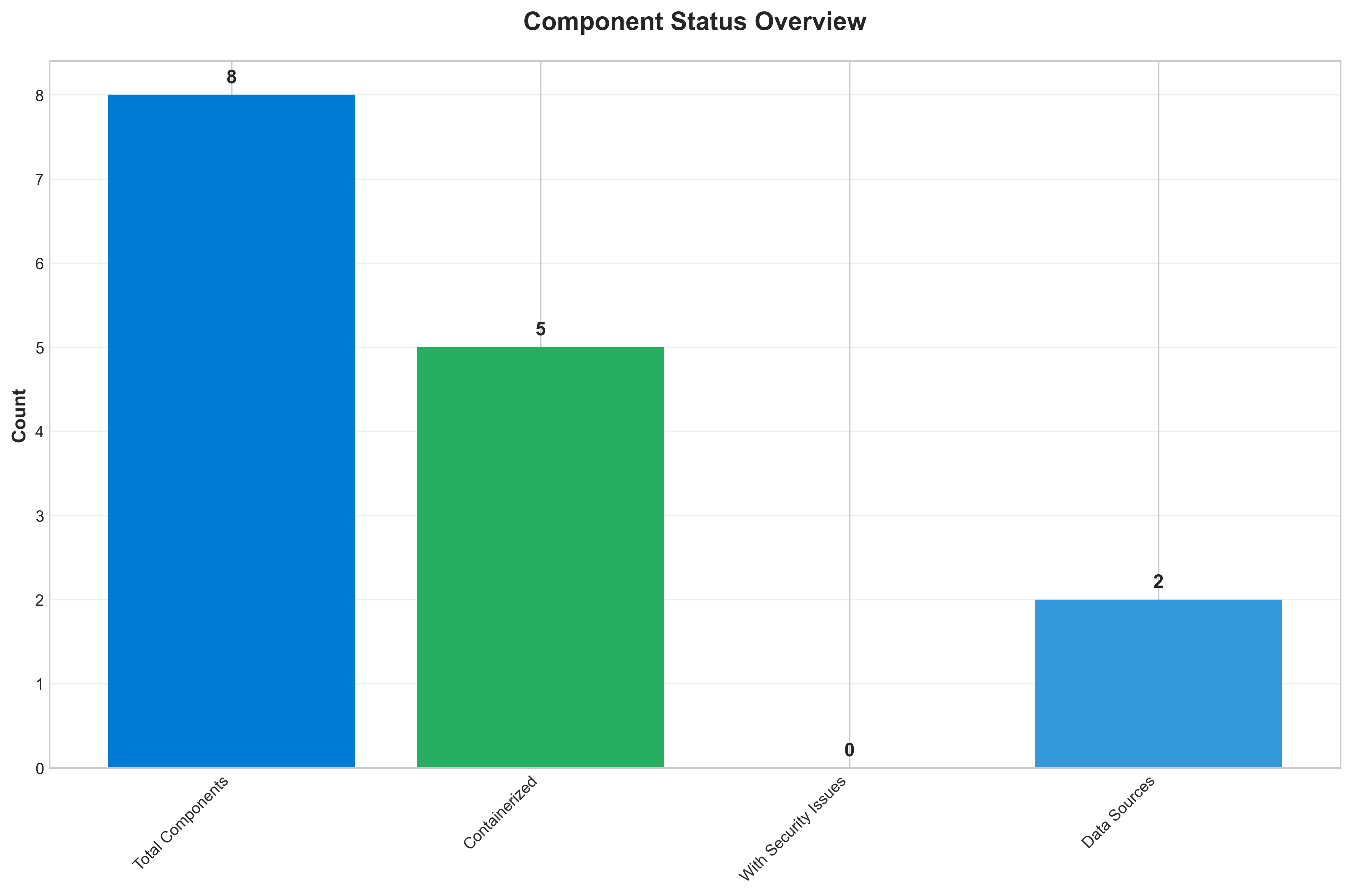
• Git History Analysis: Examination of commit patterns and development activity

Charts and Visualizations

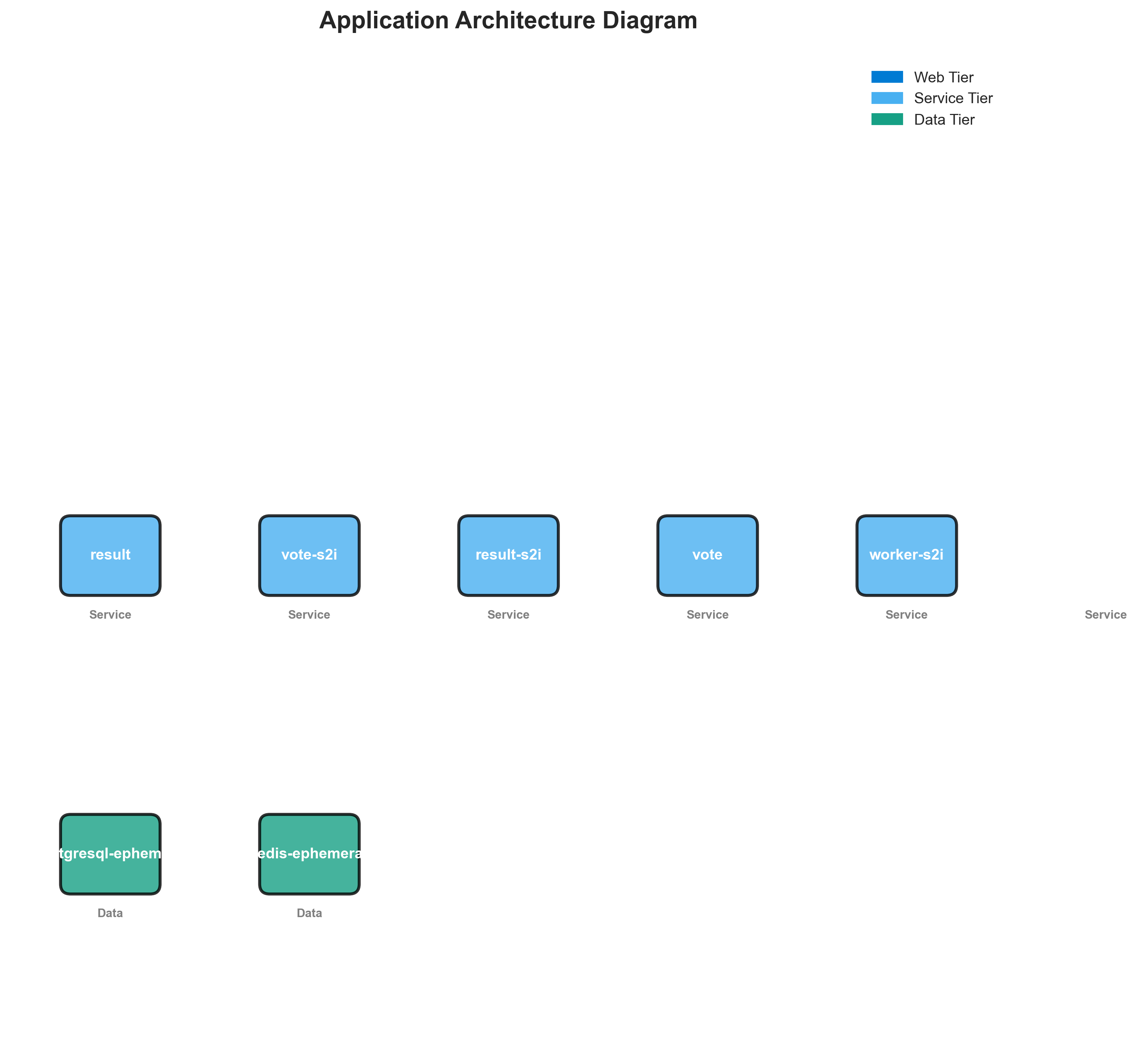
Programming Language Distribution



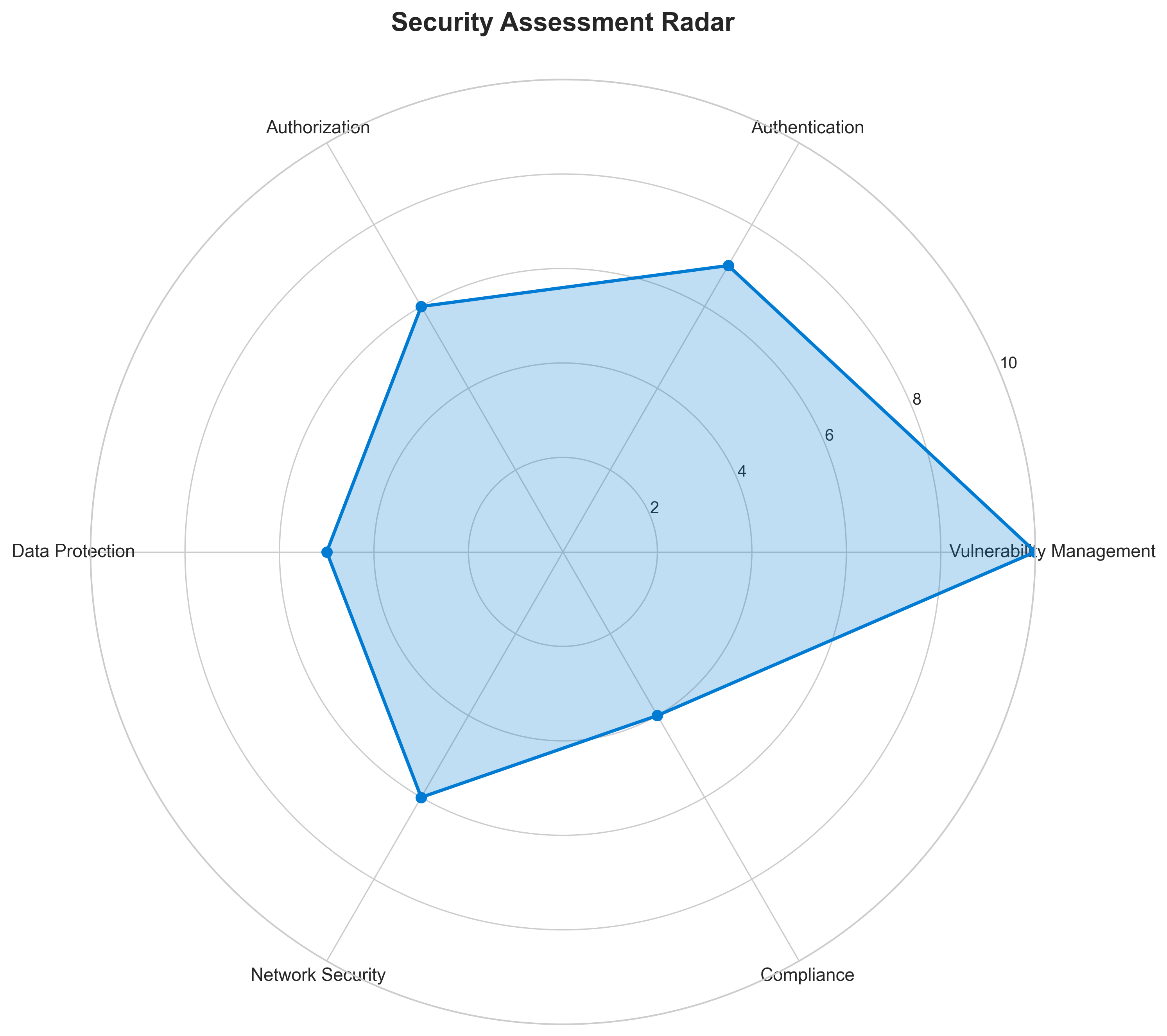
Component Status Overview



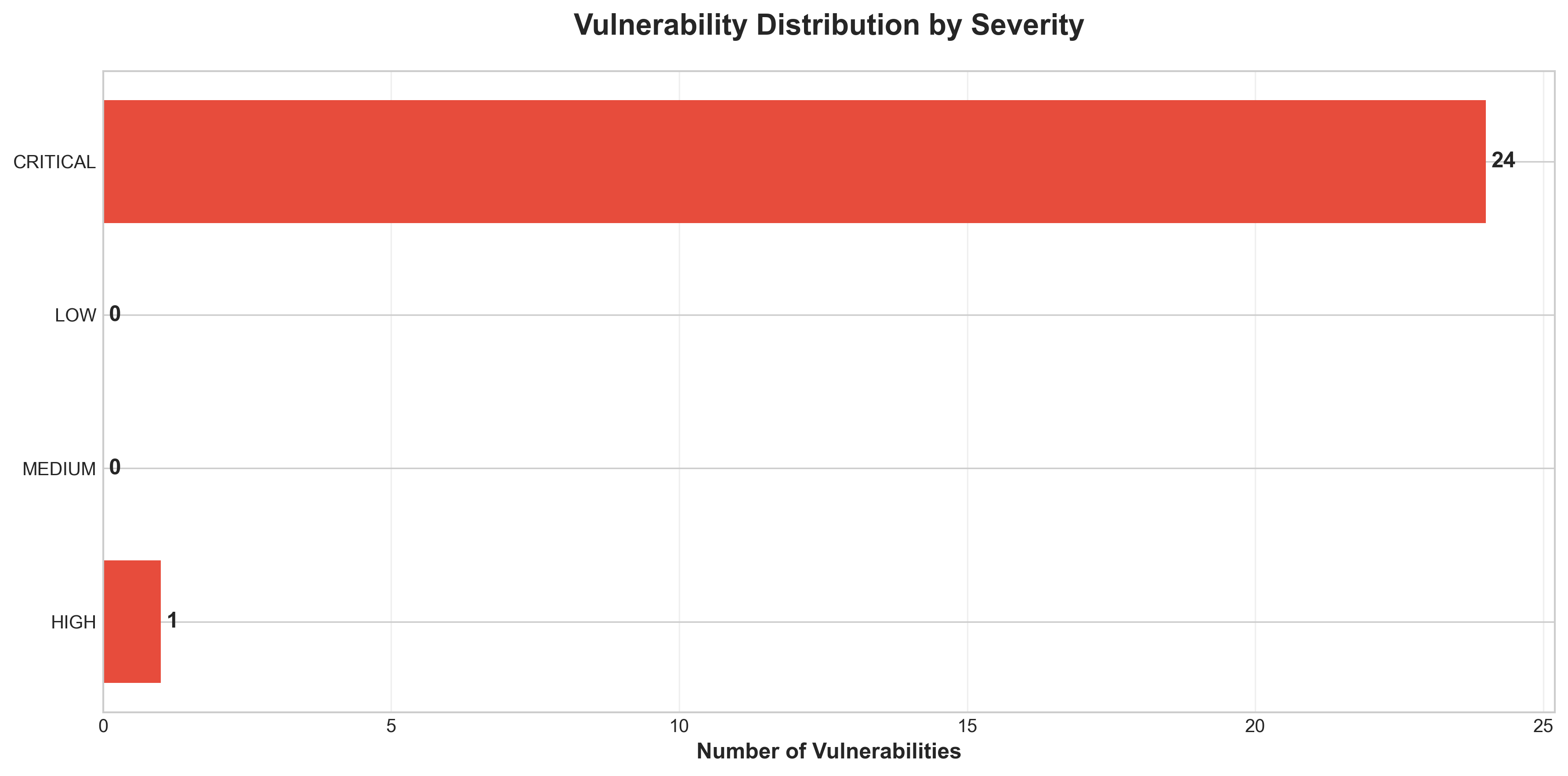
Application Architecture



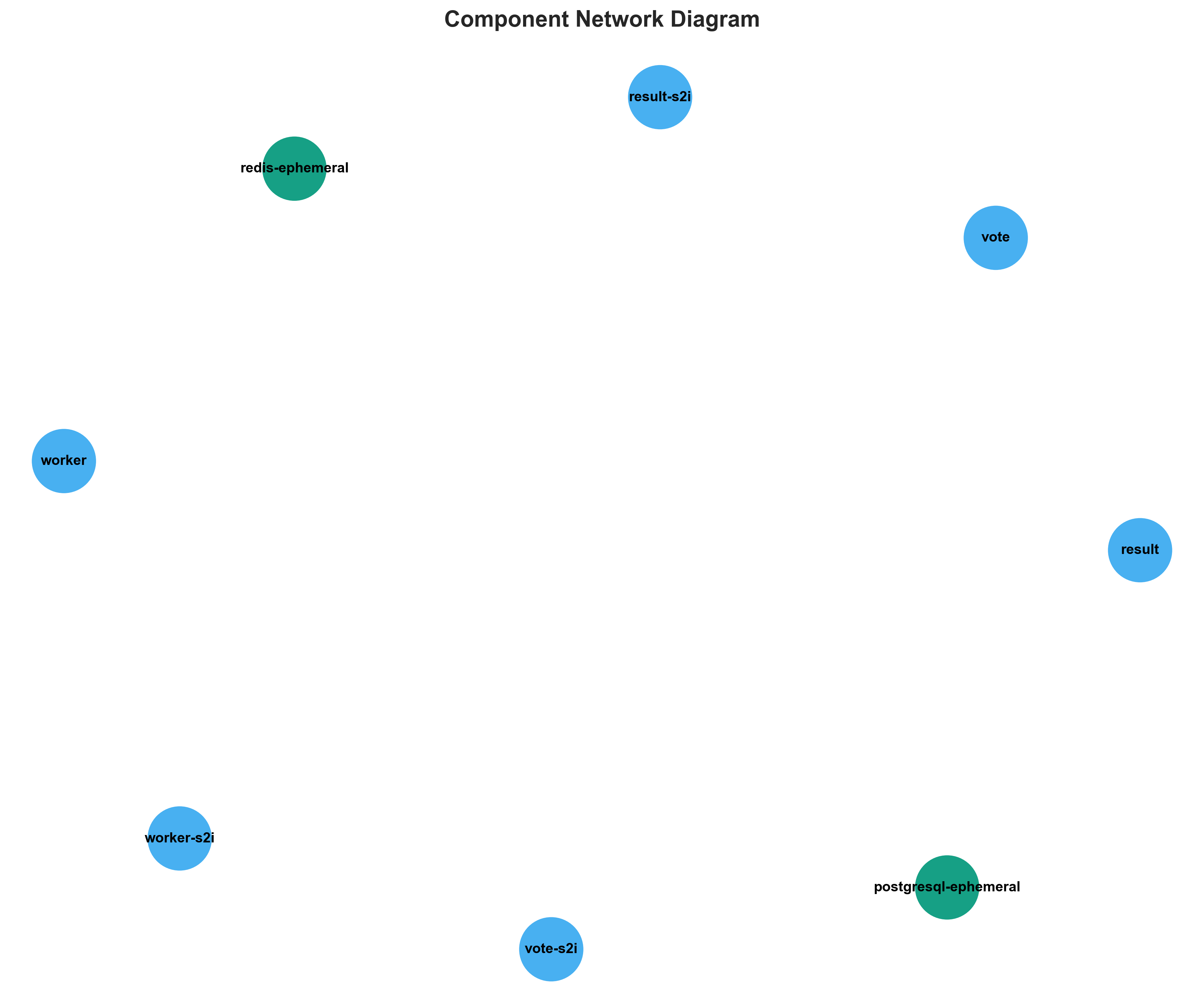
Security Assessment Radar



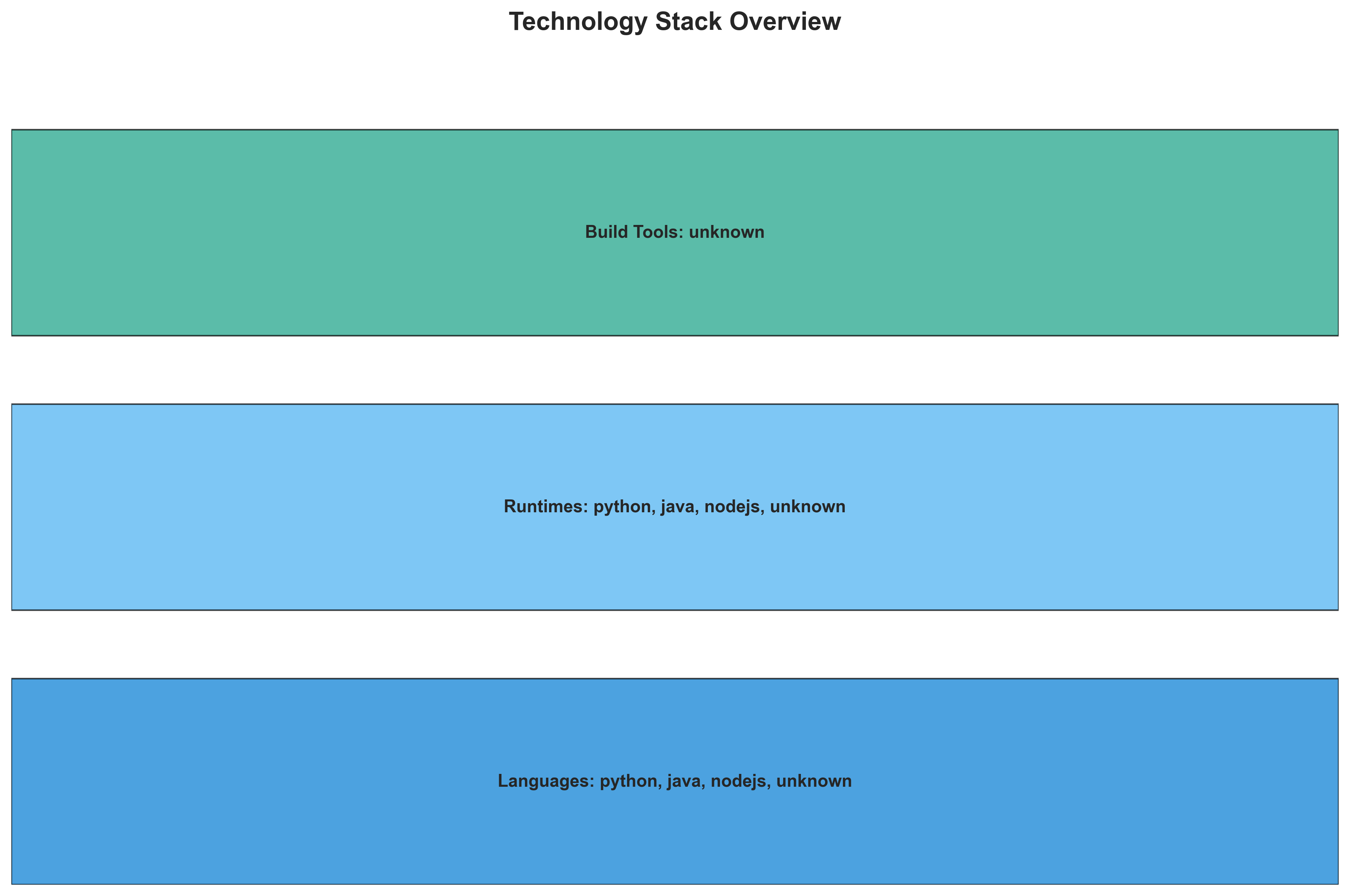
Vulnerability Analysis



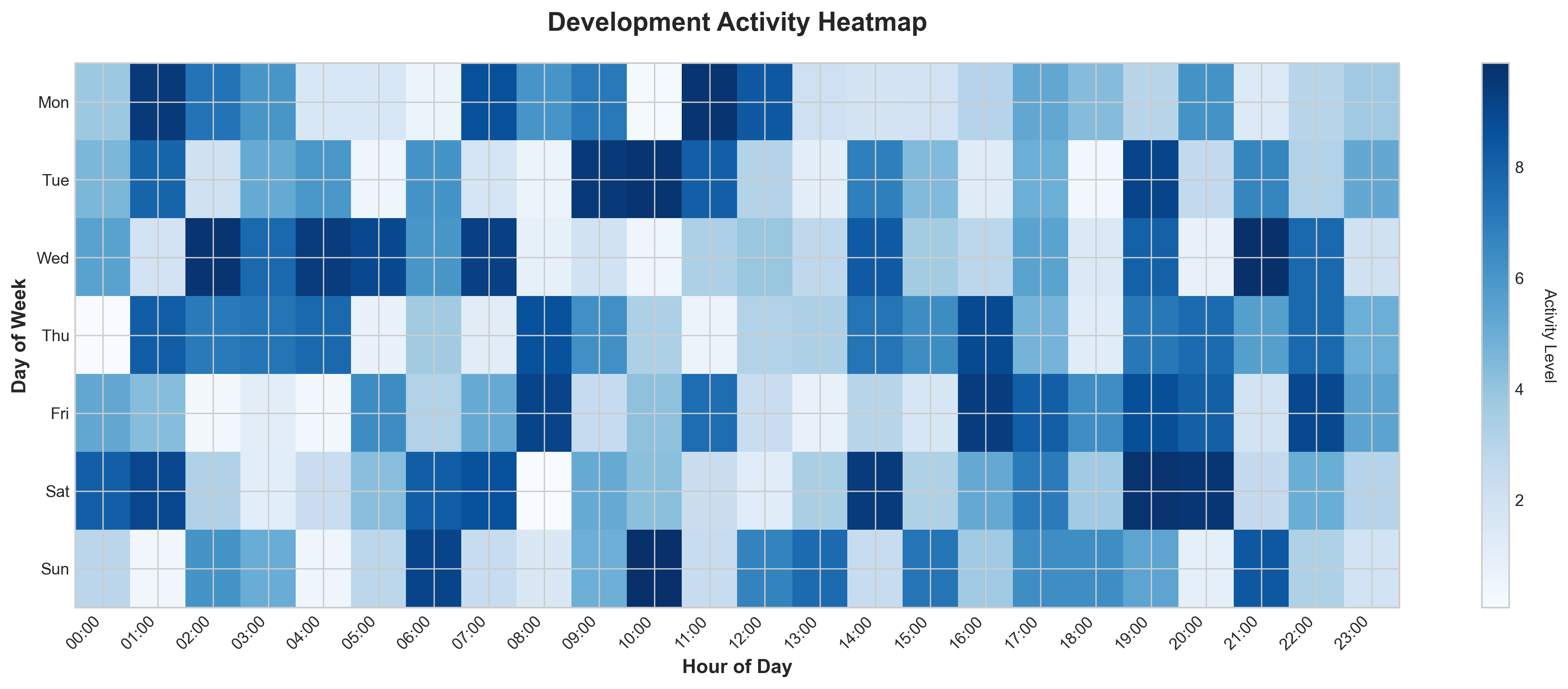
Component Network Topology



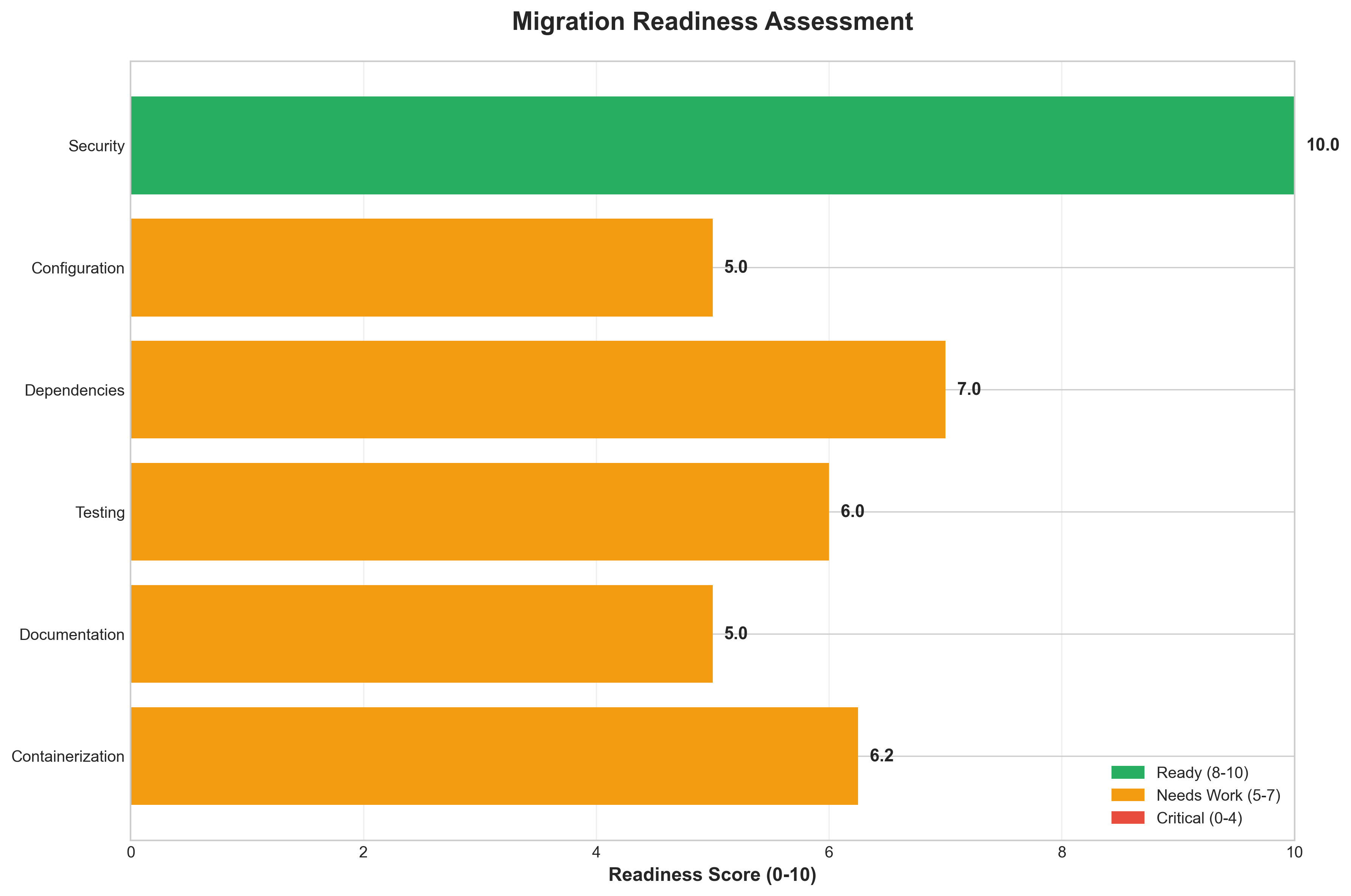
Technology Stack



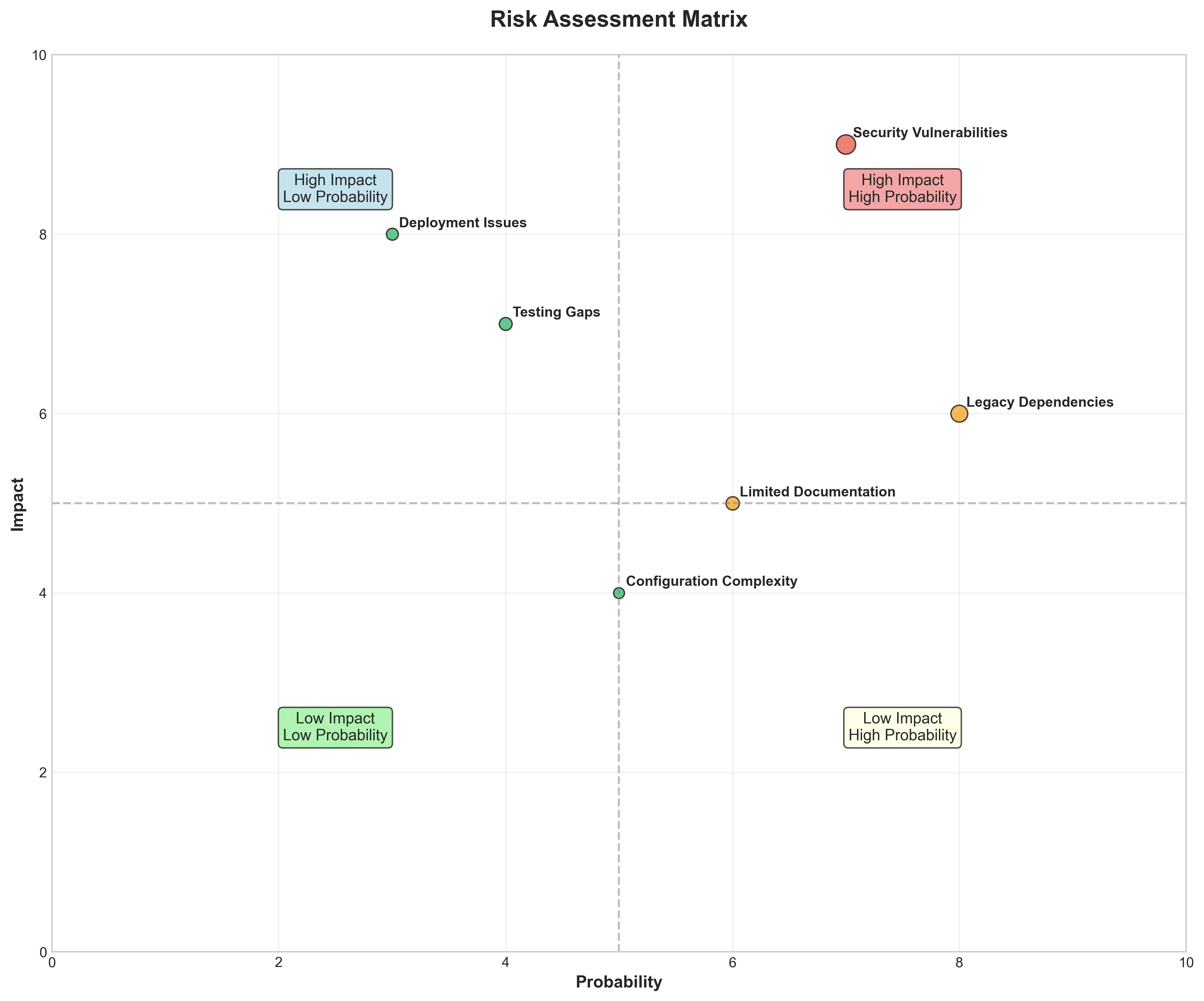
Development Activity Heatmap



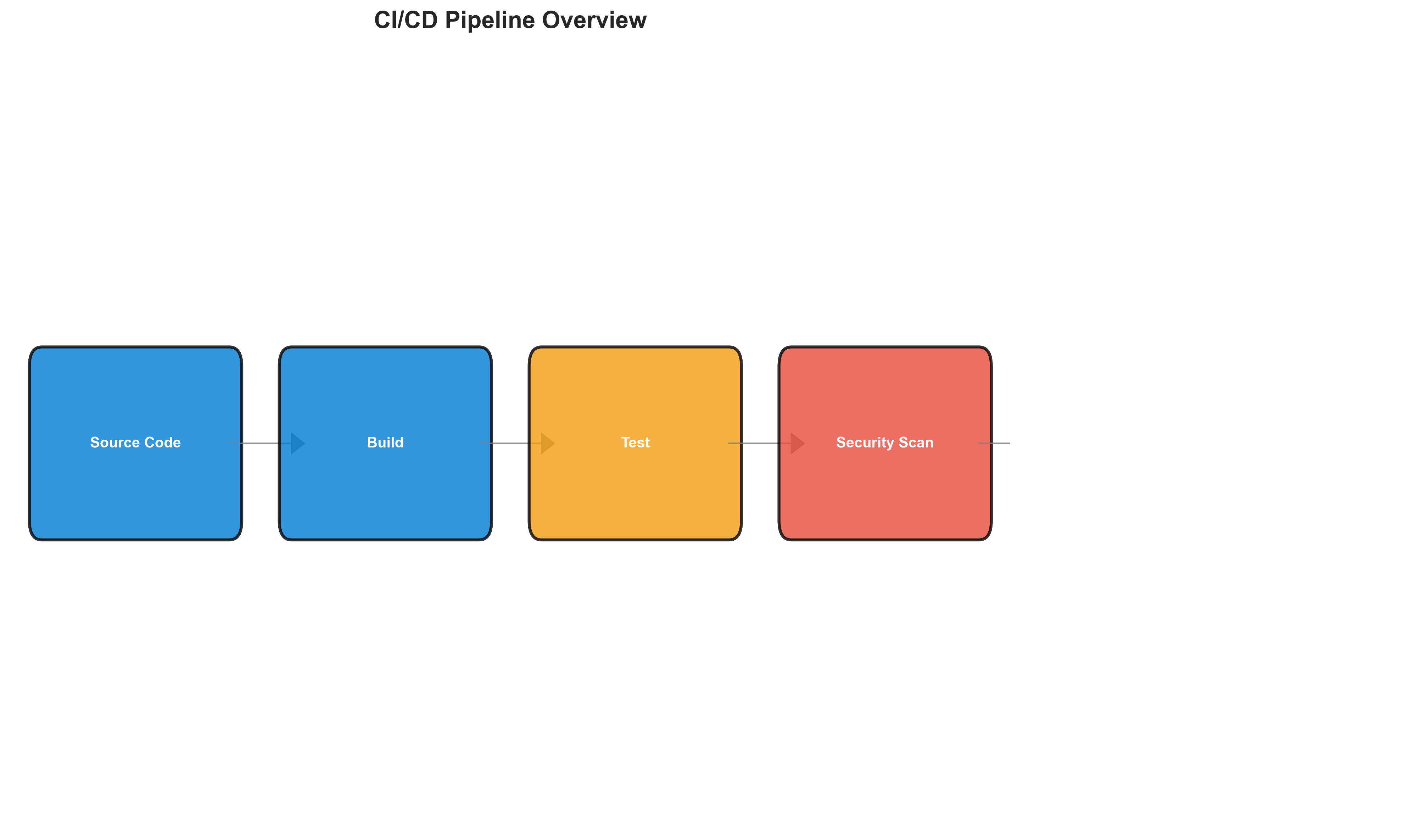
Migration Readiness Assessment



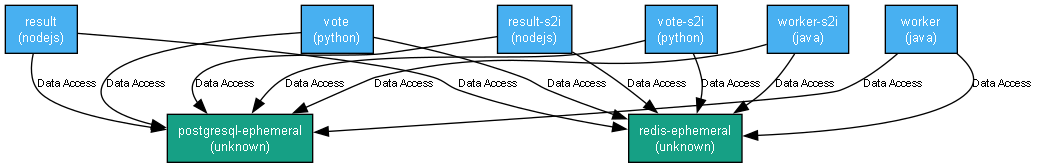
Risk Assessment Matrix



CI/CD Pipeline Overview



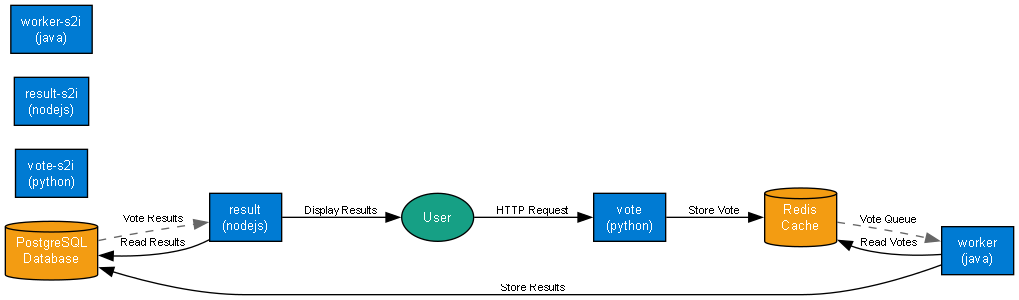
Component Relationships (Graphviz)



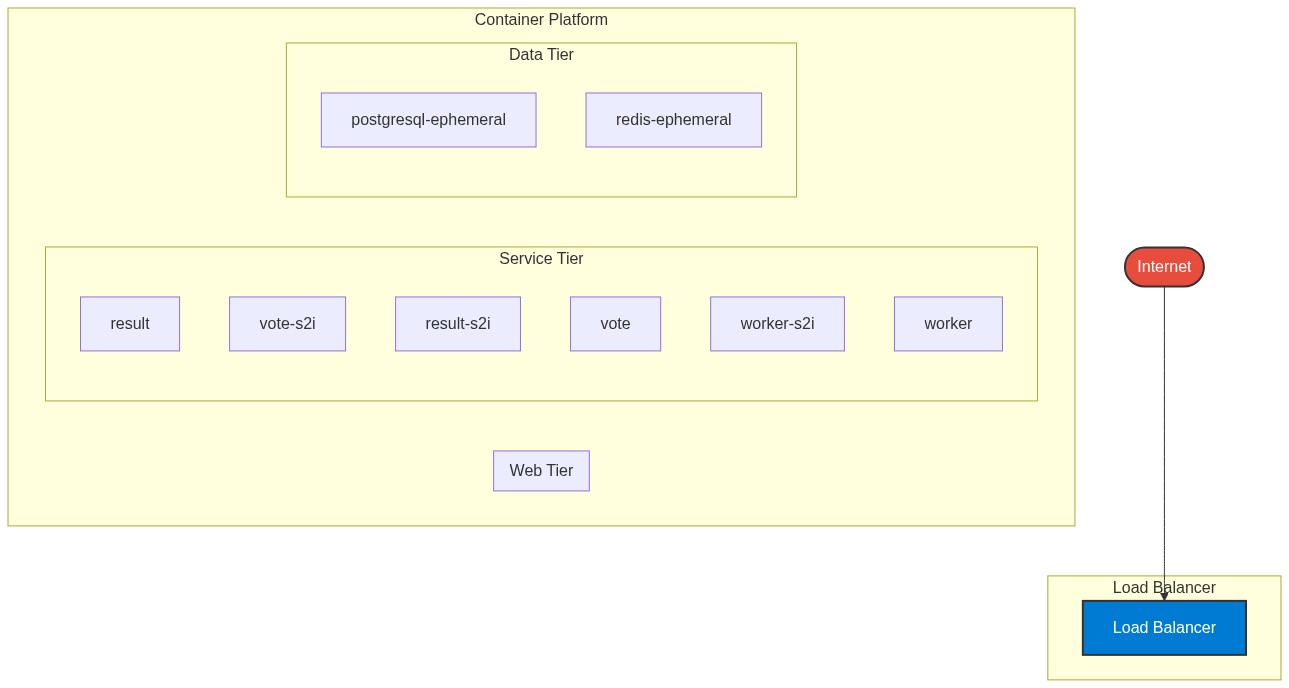
Security Flow Diagram (Mermaid)



Data Flow Diagram (Graphviz)



Deployment Architecture (Mermaid)



Risk Assessment Flow (Mermaid)



Migration Strategy (Graphviz)

