Product Requirements Document (PRD)

Product Name: VulnScope

Prepared by: Prudvi Kumar Reddy. P
Date: April 10, 2025
Version: 1.0

1. 🔍 Overview

Purpose

VulnScope is a container image vulnerability detection and management platform designed to help security engineers and DevOps teams identify, prioritize, and remediate vulnerabilities in containerized environments. As container usage scales, so does the need for a centralized, insightful, and user-friendly interface to continuously monitor thousands of images for security flaws.

Scope

This product aims to scan container images (hosted in container registries like DockerHub, ECR, GCR, etc.), identify known vulnerabilities using reliable vulnerability databases (e.g., NVD, Red Hat CVE feeds), classify their severity, and empower users to take actionable steps toward remediation.

The scope includes:

- Integrating with container registries
- Periodic and on-demand vulnerability scanning
- Presenting vulnerability insights through a clean, filterable UI
- Allowing sorting, search, and severity-based filtering
- Providing fix suggestions and remediation support

2. 🧩 Problem Statement

In cloud-native environments, container images are built and deployed at scale, often with third-party software dependencies. As a result, many organizations unknowingly ship software with critical vulnerabilities. Traditional scanning tools are fragmented, hard to scale, and provide limited visibility across the entire image inventory.

Challenges faced by users include:

- Inability to scan and monitor thousands of images simultaneously
- Lack of clarity on which vulnerabilities are most severe or exploitable
- No structured prioritization for remediation
- Tedious navigation to investigate or fix image issues

VulnScope addresses these issues by offering a centralized vulnerability scanner and explorer tailored for scale, clarity, and action.

3. 🎯 Goals and Objectives

Primary Goals

- Provide visibility into the security posture of container images across all repositories.
- Enable efficient detection of vulnerabilities using modern scanning engines.
- Prioritize vulnerabilities based on severity, exploitability, and fix availability.
- Support streamlined workflows for remediation, including manual and automated scans.

Objectives

- Achieve scan coverage of at least 10,000 images per organization with minimal performance impact.
- Offer comprehensive views of image metadata and vulnerability breakdowns.
- Reduce average time to identify and remediate a critical vulnerability.
- Equip users with actionable insights — not just data — to drive secure DevOps practices.

4. 👤 User Personas

🧑‍💻 DevOps Engineer

- Works with CI/CD and container deployment pipelines.
- Needs quick visibility into risky images and automatable ways to remediate them.

🕵️ Security Analyst

- Reviews system and software security posture.
- Investigates vulnerability trends, exposure, and CVE impact.
- Requires detailed filtering and CVE drill-down capabilities.

🛠️ Platform Engineer

- Owns platform-wide tooling and integrations.
- Needs APIs and integrations with existing scanning systems.
- Ensures that image scanning is integrated into DevSecOps workflows.

5. 🧪 User Stories

ID	User Story	Priority	Acceptance Criteria
US1	As a user, I want to see a high-level overview of vulnerability status across all images	High	Dashboard with total images, vulnerable images, top severity chart
US2	As a user, I want to search and filter images by severity, fixability, repository, or date	High	Functional filters and search bar working on list view
US3	As a user, I want to drill down into a single image to view vulnerabilities and fix suggestions	High	Image details page with CVE metadata, severity, fixes
US4	As a user, I want to trigger a manual scan for a specific image	Medium	"Scan Now" button triggers background scan
US5	As a user, I want to export vulnerability reports to CSV or PDF formats	Medium	Export feature with customizable report content

US6	As a user, I want to know which vulnerabilities have available fixes	High	Fixability column in vulnerability list
US7	As a user, I want to sort images by vulnerability count, severity, or scan date	Medium	Sort icons and functionality in table headers

6. ✨ Features and Requirements

6.1 Dashboard

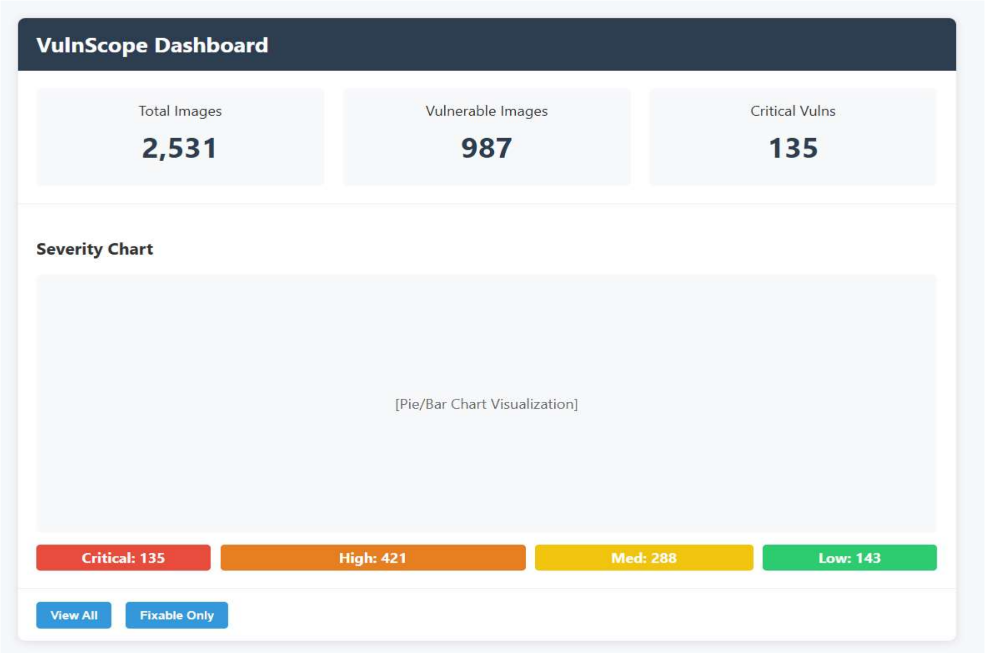
A single-pane overview that provides actionable metrics at a glance.

Features:

- KPIs: Total Images, Vulnerable Images, Critical/High Vulnerabilities
- Severity Distribution Chart (Bar or Pie Chart)
- Recently Scanned / Most Critical Images
- Alert banner for images with exploitable critical vulnerabilities

User Benefits:

- Understand system-wide exposure immediately
- Act fast on critical alerts
- Stay informed with updated scanning trends



VulnScope Dashboard

Total Images
2,531

Vulnerable Images
987

Critical Vulns
135

Severity Chart

[Pie/Bar Chart Visualization]

6.2 Container Image Inventory View

Features:

- Tabular display of all images with sortable columns:
  - Image Name
  - Repository
  - Vulnerability Count
  - Highest Severity
  - Last Scanned
- Filters (multi-select):
  - Severity (Critical, High, Medium, Low)
  - Fixable Only
  - Repository Name
  - Last Scanned Range

User Benefits:

- Locate problematic images quickly
- Prioritize scanning and remediation workload
- Manage large inventories with ease

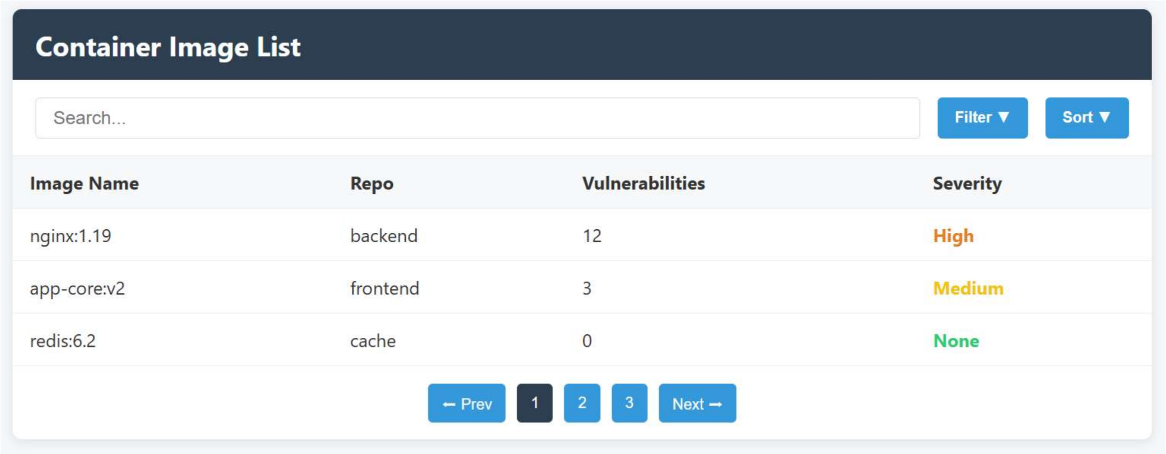


Image Name	Repo	Vulnerabilities	Severity
nginx:1.19	backend	12	High
app-core:v2	frontend	3	Medium
redis:6.2	cache	0	None

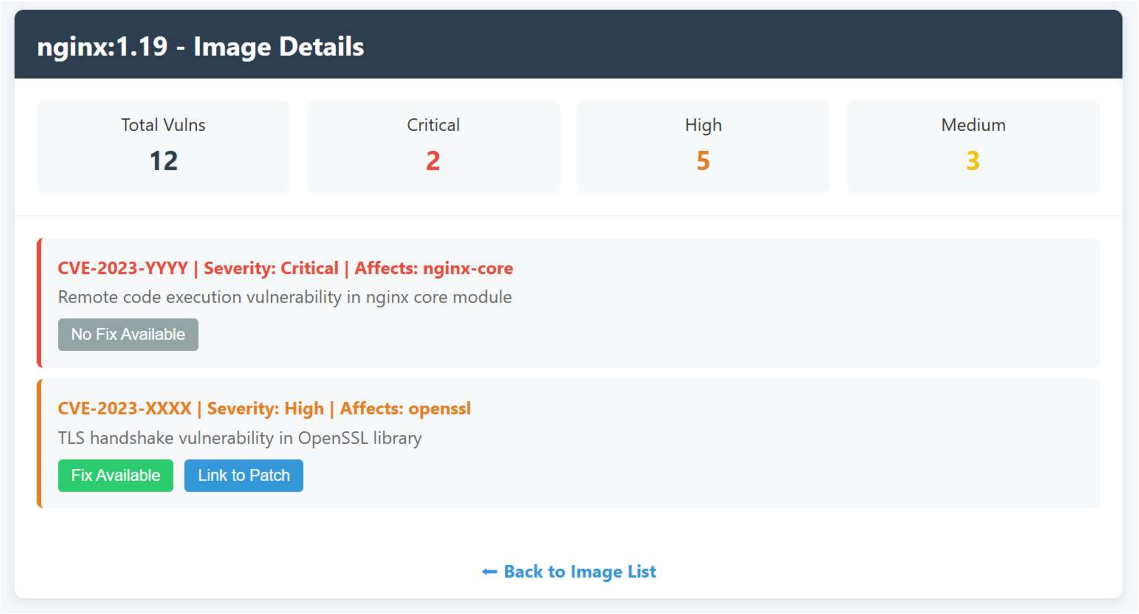
6.3 Image Detail View

Features:

- Image Metadata: Name, Tag, Digest, Repo, Scan Time
- Vulnerability Table (sortable/filterable):
  - CVE ID
  - Package Name & Version
  - Severity Level
  - Fix Version (if available)
  - Description & References
- Actions:
  - Trigger Manual Rescan
  - Mark CVE as reviewed or accepted
  - View Dockerfile layers (optional)

User Benefits:

- Get deep visibility into individual image issues
- Take immediate action (e.g., rescan or export report)
- Access reference materials for each CVE



nginx:1.19 - Image Details

Total Vulns	Critical	High	Medium
12	2	5	3

CVE-2023-YYYY | Severity: Critical | Affects: nginx-core

Remote code execution vulnerability in nginx core module

CVE-2023-XXXX | Severity: High | Affects: openssl

TLS handshake vulnerability in OpenSSL library

6.4 Vulnerability Fix Guidance

Features:

- Fix version suggestions for affected packages
- Mark vulnerabilities as fixed post-update (with verification)
- CI/CD guidance to rebuild image with updated dependencies
- Optional integration with issue trackers (e.g., Jira)

User Benefits:

- Remediate vulnerabilities faster
- Avoid manual digging for upgrade paths
- Integrate into development sprint cycles

6.5 Exporting and Reporting

Features:

- Export reports (CSV/PDF) from:
  - Dashboard Summary
  - Image Table
  - Individual Image Detail

- Select vulnerability fields to include
- Apply filters before export (e.g., critical only)

User Benefits:

- Share reports with compliance/security teams
- Maintain records for audits
- Use offline or in presentations

7. 📊 Success Metrics

To validate the impact and usability of VulnScope, we'll track the following KPIs:

Metric	Target
% of critical vulnerabilities resolved within 7 days	70%+
Avg time from scan to fix suggestion	< 1 minute
Avg user satisfaction score on UI/UX	4.5/5
Scan coverage across 10,000+ images	> 95%
Number of export/download events per week	> 100
Active users per month	> 500

These metrics will be monitored post-launch and will inform product iteration decisions.

8. ☑ Non-Functional Requirements

Category	Requirement
Performance	Must support 10,000+ images with <2s load time per view
Security	Encrypted storage, Role-Based Access Control, Audit logging
Scalability	Horizontally scalable scan engine using Kubernetes
Availability	99.9% uptime, redundant deployments
Compliance	Alignment with ISO 27001, SOC2, and industry standards
Responsiveness	Web UI should support desktops, tablets, and smaller laptop screens

9. 🚀 Development Action Items

Backend Engineering:

- Integrate with Trivy/Grype scanning engines
- Set up scan job orchestration (Kubernetes CronJobs or Celery)
- Develop REST APIs for:
  - Image ingestion
  - Vulnerability fetch
  - Scan initiation
  - Report export

Frontend Engineering:

- Build UI using React + Tailwind
- Implement dashboard, list view, and detail page
- Add filter/sort/search functionality
- Implement CSV/PDF export using client-side rendering or server fetch

DevOps & Infra:

- Setup container registry integrations
- Configure secure OAuth2-based authentication
- Implement CI/CD pipelines for backend/frontend deployments
- Configure logging, alerting, and audit trail collection

Security:

- Enforce API security with JWT
- Implement RBAC (Admin, Analyst, Viewer roles)
- Ensure all third-party dependencies are vulnerability-free

10. 🧪 QA and Testing

Testing Strategies:

- **Unit Testing**: All API endpoints, CVE mapping logic, and frontend components

- **Integration Testing**: Registry connectors, scan result ingestion
- **UI Testing**: Table rendering, filters, forms, pagination, export
- **Load Testing**: Simulate large image datasets
- **Security Testing**: Penetration tests and token validation tests

11. 🧾 Conclusion

VulnScope is designed to address a critical gap in container security — providing scalable, actionable, and intelligent visibility into container image vulnerabilities. As organizations continue to scale their container-based deployments, security cannot be an afterthought. VulnScope empowers DevOps, Security Analysts, and Platform Engineers with the tools they need to detect, prioritize, and fix vulnerabilities across thousands of images, all through a unified and user-friendly interface.

By combining real-time scanning, fix recommendations, intuitive dashboards, and exportable reporting, VulnScope doesn't just highlight problems — it guides users toward resolution. This PRD outlines a roadmap to deliver a high-impact, high-utility solution that balances security best practices with performance and ease of use.

With strong cross-functional collaboration between design, engineering, security, and DevOps, VulnScope can rapidly evolve into a core pillar of enterprise container security strategy. The next steps include finalizing wireframes, initiating design sprints, and executing on the outlined development action items to bring this vision to life.