# Agentic SI Container Vulnerability Analysis

*From Vulnerability to Exploitability-Based Decision Making*

IntelliGenesis LLC
WHERE INTELLIGENCE BEGINS
info@intelligenesisllc.com
intelligenesisllc.com

## The Problem: Detection Does Not Equal Risk

Modern containerized environments generate an overwhelming volume of vulnerability data. Traditional security tools identify known vulnerabilities (CVEs) across container images and software dependencies, often producing thousands of findings per scan.

**However, these tools stop at detection. They do not determine:**

- Whether a vulnerability is reachable in the application
- Whether it can be exploited in the given environment
- Whether it meaningfully impacts operational risk

As a result, security teams are left with a critical gap between identifying vulnerabilities and understanding real risk.

**To bridge that gap, analysts must manually:**

- Research CVEs across multiple data sources
- Trace dependency relationships through SBOMs
- Evaluate code paths and configuration context
- Document justification for prioritization and remediation

**This process is:**

- Time-intensive
- Difficult to scale
- Highly dependent on analyst expertise

***In high-volume environments, the bottleneck is not detection, it is decision-making.***

## The Shift: From Vulnerability Lists to Risk Determination

DryDock addresses this gap by shifting vulnerability management from enumeration to evaluation.

**Instead of asking:**
**"What vulnerabilities exist?"**

**DryDock answers:**
**"Which vulnerabilities actually matter?"**

This shift changes the role of security tooling from producing data to enabling decision-ready outputs.

## What DryDock Does

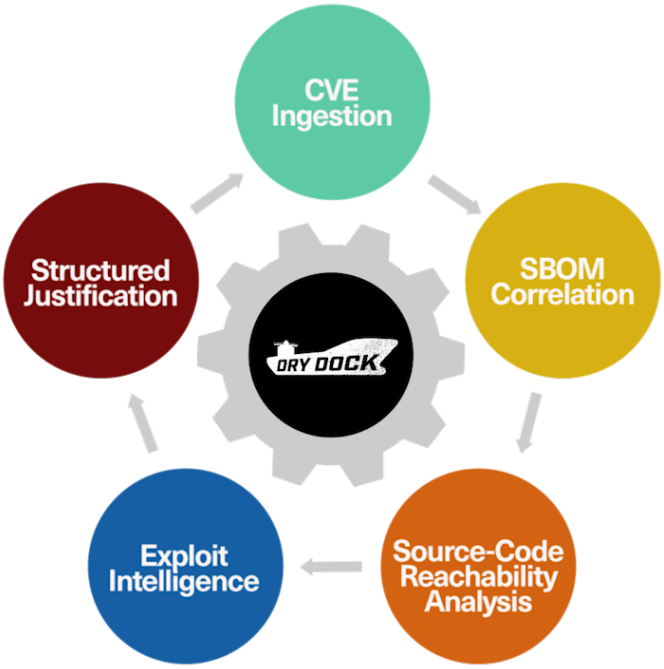
**DryDock is an SI-driven platform that performs:**

- Vulnerability scanning
- SBOM generation and analysis
- Contextual exploitability assessment

It integrates these capabilities into a single workflow that evaluates vulnerabilities within the actual application and dependency context.

**Rather than generating large, unfiltered lists, DryDock produces structured outputs that indicate:**

- Whether a vulnerability is exploitable
- Why that determination was made
- What components and dependencies are involved




**DryDock identifies which vulnerabilities are *actually exploitable* in your containerized applications.**

## What Makes DryDock Different

Most tools in the container security ecosystem focus on detection or visibility. DryDock focuses on decision-making.

**Traditional Approaches:**

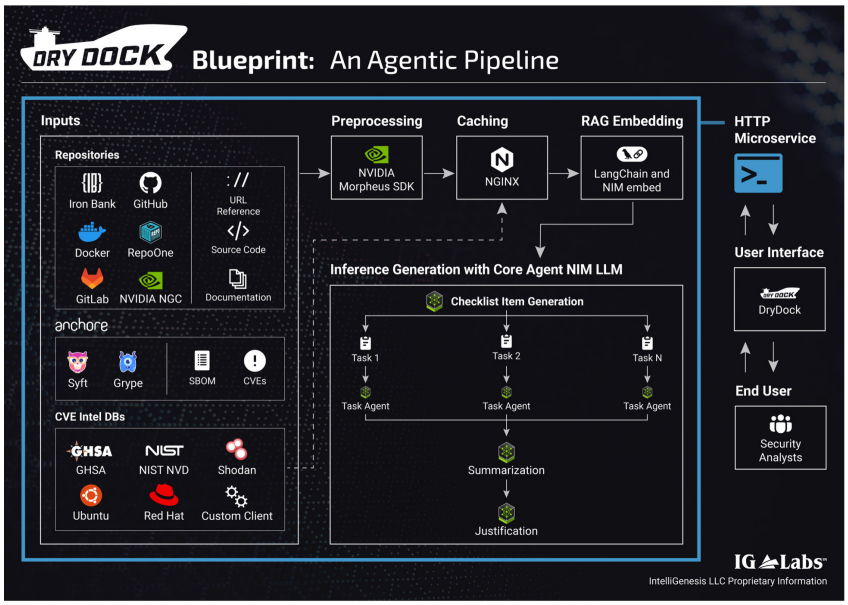
- Identify vulnerabilities
- Provide severity scores (CVSS)
- Require manual triage

**DryDock Approach**

- Evaluates real-world exploitability
- Provides context-aware analysis
- Produces evidence-backed conclusions

***This distinction is critical. Security teams do not need more alerts.They need fewer, better decisions.***

# How it Works




*Scan container images and analyze software components*

*Generate and evaluate SBOM and dependency relationships*

*Assess exploitability based on application context*

*Provide evidence-backed results for prioritization*

**DryDock uses an agentic SI pipeline to *automate the analysis process* typically performed by security analysts.**

### At a high level, the system:

1. **Ingests container images and software metadata**
   - Extracts dependencies and builds SBOMs
2. **Correlates vulnerability intelligence**
   - Pulls CVE data from authoritative sources (NVD, GHSA, vendor advisories)
3. **Analyzes application context**
   - Evaluates how vulnerable components are used
   - Traces dependency paths
   - Assesses code path reachability
4. **Determines exploitability**
   - Applies contextual reasoning to determine whether a CVE is realistically exploitable
5. **Generates explainable outputs**
   - Provides justification, evidence, and structured reporting (including VEX-style outputs)

***This transforms a multi-hour (or multi-day) manual workflow into an automated, repeatable process.***

IntelliGenesis
WHERE INTELLIGENCE BEGINS

**DryDock directly addresses the *primary bottleneck* in modern security workflows: manual triage.**

## Operational Impact

### Key Improvements

- **Reduced analysis time**
  - Automated exploitability assessment replaces manual research and validation
- **Lower false positive burden**
  - Focus shifts from theoretical vulnerabilities to actionable risk
- **Improved prioritization**
  - Teams can act on what matters instead of sorting through noise
- **Increased analyst efficiency**
  - Analysts spend less time gathering data and more time making decisions
- **Faster security outcomes**
  - Vulnerability identification → prioritization → remediation happens faster

## Where it Fits

### DryDock is designed for environments where:

- Containerized applications are widely used
- Vulnerability volume exceeds manual capacity
- Security teams operate within CI/CD pipelines
- SBOMs and software supply chain security are already in scope

### Primary Users

- DevSecOps teams
- Application security teams
- Platform security teams

### Typical Environments

- Kubernetes-based infrastructure
- Cloud-native applications
- Enterprise DevSecOps pipelines

## The Bottom Line

**Vulnerability scanners answer:**
**"What is there?"**

**DryDock answers:**
**"What matters?"**

DRY DOCK

**By automating exploitability analysis and providing explainable, decision-ready outputs, DryDock enables organizations to move from data overload to focused risk management.**

**Contact us to learn more or schedule a walkthrough of DryDock.**

info@intelligenesisllc.com | www.intelligenesisllc.com