DRY DOCK

USE CASE:

From 1,000+ Vulnerabilities to *What Actually Matters*

THE SITUATION:

A DevSecOps team scans a container image before deployment.

1,200+

VULNERABILITIES DETECTED*

*This is a normal result in modern environments and encompasses base images, libraries, and dependencies.

THE PROBLEM:

Traditional tools provide visibility, *not answers*.

The team must manually:

- Trace dependency relationships
- Evaluate code path reachability
- Determine if vulnerabilities are actually exploitable
- Document findings for prioritization and remediation

THE COST:

The process is:

- Time-intensive
- Difficult to scale
- Dependent on analyst expertise

As a result, teams often:

- Delay remediation
- Prioritize incorrectly
- Spend time on issues that don't matter

HOW DRYDOCK WORKS:

DryDock ingests the same scan results and:

01
MAPS
dependencies and relationships

02
ANALYZES
application context

03
EVALUATES
real world exploitability

04
DELIVERS
evidence-backed conclusions

Traditional Tools Answer: "What Is There?"

DryDock Answers: "What Matters?"

> **Instead of a list that requires further investigation, the team receives a result that is ready to act on.**
>
> - *A focused set of vulnerabilities*
> - *A clear explanation of why each one matters*
> - *Supporting context for confident decision-making*

THE OUTCOME:

WITHOUT DRYDOCK:

- 1,200+ vulnerabilities
- Manual investigation
- Unclear prioritization

WITH DRYDOCK:

- A small set of actionable risks
- Clear justification for each decision
- Faster, confident remediation

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

intelligenesisllc.com info@intelligenesisllc.com