



Secure Every Link In the Software Supply Chain

With the Oligo Application Defense Platform

Components are added and changed in every step of the software development life cycle—and every step creates new supply chain complexity and potentially introduces new exploitable vulnerabilities.

The Oligo Platform addresses the crucial last link in the software supply chain: applications at runtime.

Using Oligo, organizations can see every component of every application they build, buy, or use—stopping supply chain blind spots and allowing the complete supply chain to be observed and secured.



Scan All Your Applications — No Source Code Required

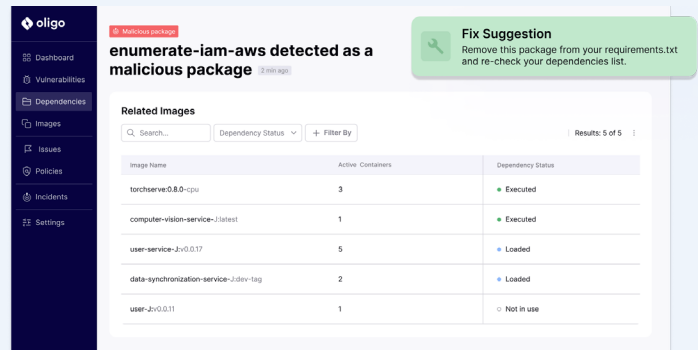
Organizations today must ensure that every application they use is free from compromises — whether it was built in-house or elsewhere.

Tools that work at an earlier stage in the SDLC (like repository and pipeline scanners) can only see first-party application source code. Oligo works differently: positioned at the end of the supply chain, when applications are actually running, the Oligo Application Defense Platform is the only solution on the market that can observe and analyze every link in the supply chain.

The Oligo Application Defense Platform makes it easy to understand and contextualize supply chain risk, without depending on supplier SBOMs—which may be incomplete or lacking context, and fall out of date rapidly. With Oligo, knowing what's inside third-party applications is as simple as taking a direct look inside.

Stop Malicious Package Attacks

Most vulnerabilities will never be exploited—but malicious packages start doing damage as soon as they find their way to your applications. Oligo works differently from other supply chain security products, detecting behavioral anomalies at runtime and identifying malicious behavior in your code.

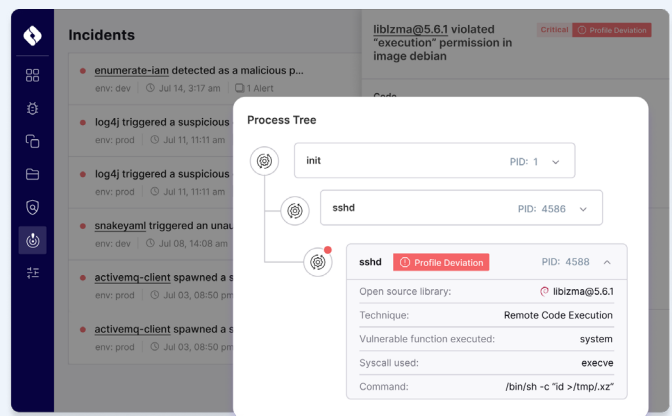


Malicious package
enumerate-iam-aws detected as a malicious package 5 min ago

Fix Suggestion
 Remove this package from your requirements.txt and re-check your dependencies list.

Related Images

Image Name	Active Containers	Dependency Status
torchserve:0.8.0-cpu	3	Executed
computer-vision-service-latest	1	Executed
user-service-jv:0.0.17	5	Loaded
data-synchronization-service-j-dev-tag	2	Loaded
user-jv:0.0.11	1	Not in use



Incidents

- enumerate-iam detected as a malicious p... env: dev Jul 14, 3:17 am 1 Alert
- log4j triggered a suspicious... env: prod Jul 11, 11:11 am
- log4j triggered a suspicious... env: prod Jul 11, 11:11 am
- snakeyaml triggered an unau... env: dev Jul 08, 14:08 am
- activemq-client spawned a s... env: prod Jul 03, 08:50 pm
- activemq-client spawned a s... env: prod Jul 03, 08:50 pm

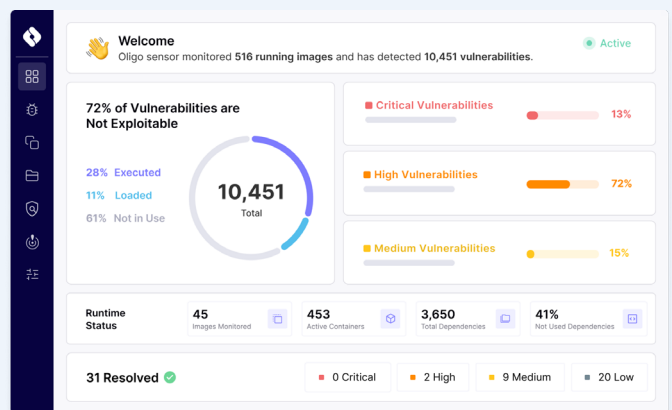
Process Tree

```

  graph TD
    init[init PID: 1] --> sshd[sshd PID: 4586]
    sshd --> sshd[sshd Profile Deviation PID: 4588]
  
```

sshd Profile Deviation PID: 4588

- Open source library: liblzma@5.6.1
- Technique: Remote Code Execution
- Vulnerable function executed: system
- Syscall used: execve
- Command: /bin/sh -c "id >/tmp/.xz"



Welcome
 Oligo sensor monitored 516 running images and has detected 10,451 vulnerabilities. Active

72% of Vulnerabilities are Not Exploitable

28% Executed
 11% Loaded
 61% Not in Use

10,451 Total

Vulnerability Breakdown:

- Critical Vulnerabilities: 13%
- High Vulnerabilities: 72%
- Medium Vulnerabilities: 15%

Runtime Status: 45 Images Monitored, 453 Active Containers, 3,650 Total Dependencies, 41% Not Used Dependencies

31 Resolved

0 Critical, 2 High, 9 Medium, 20 Low



Only Oligo

The Oligo Application Defense Platform works differently because it's built differently. Using patent-pending technology to take eBPF observability to unprecedented depths, Oligo monitors applications at the library and function level and does what no other product on the market can do:

Benefits	Other Supply Chain Solutions	Oligo
Ultra-fast deployment - Deploy in minutes, see value within 48 hours	✗	✓
Contextualize exploitability: Is the vulnerable library loaded? Is it communicating with the network? Is the vulnerable function called?	✗	✓
Full Dynamic BOM and VEX: Detect which libraries and functions are loaded, running, and executed	✗	✓
Unprecedented visibility: see contextual runtime behavior at the function level	✗	✓
Prove CVE exploitability to stakeholders	✗	✓
Prioritize fixes based on exploitability	✗	✓
Detect anomalies in all application code: open source, proprietary third-party, first-party	✗	✓
Uncover undisclosed or pre-disclosure security flaws and breaches in progress	✗	✓
Find hidden code and vulnerabilities in packaged, compiled apps	✗	✓

Oligo In Action: The XZ Backdoor

The XZ backdoor proved that supply chain threats can come from highly sophisticated actors leveraging unusual techniques to insert malicious code. The biggest supply chain protection tools in the world all failed to detect this backdoor—but at Oligo, we knew instantly that it had not been exploited in any of our customers' environments.

How could we tell? Because Oligo's detection rules could identify this indicator of compromise, any attempted exploitation would have triggered an instant Oligo alert due to the unusual way the backdoor made the library behave. Oligo could detect this type of exploitation even when the backdoor was an "unknown" threat, before its discovery or announcement.

