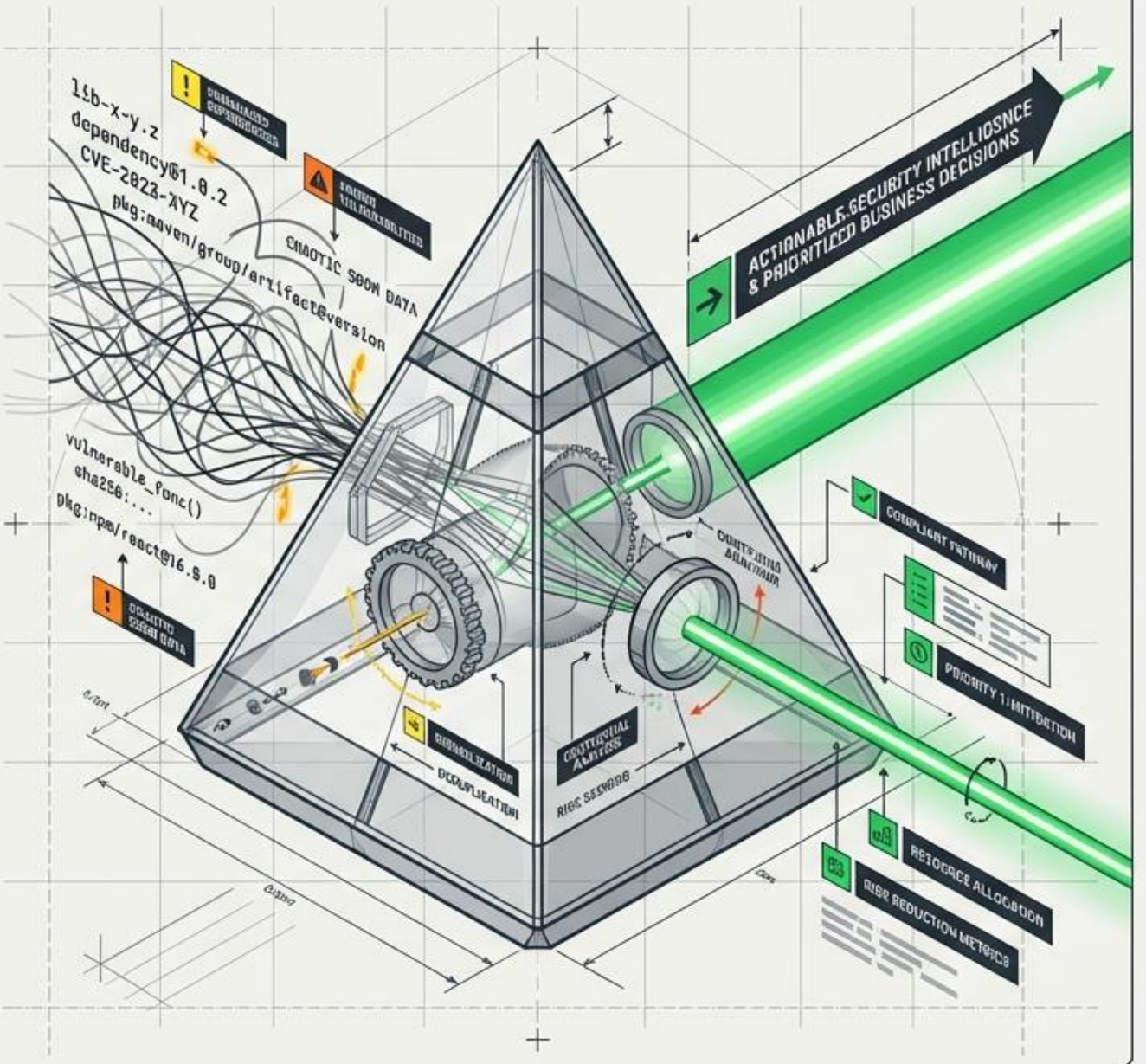
Stardust: From Component Inventory to Actionable Security

The Mandate: Cyber Resilience Act (CRA) Compliance.

The Reality: Transforming raw SBOM and CVE data into strategic, prioritized business decisions.

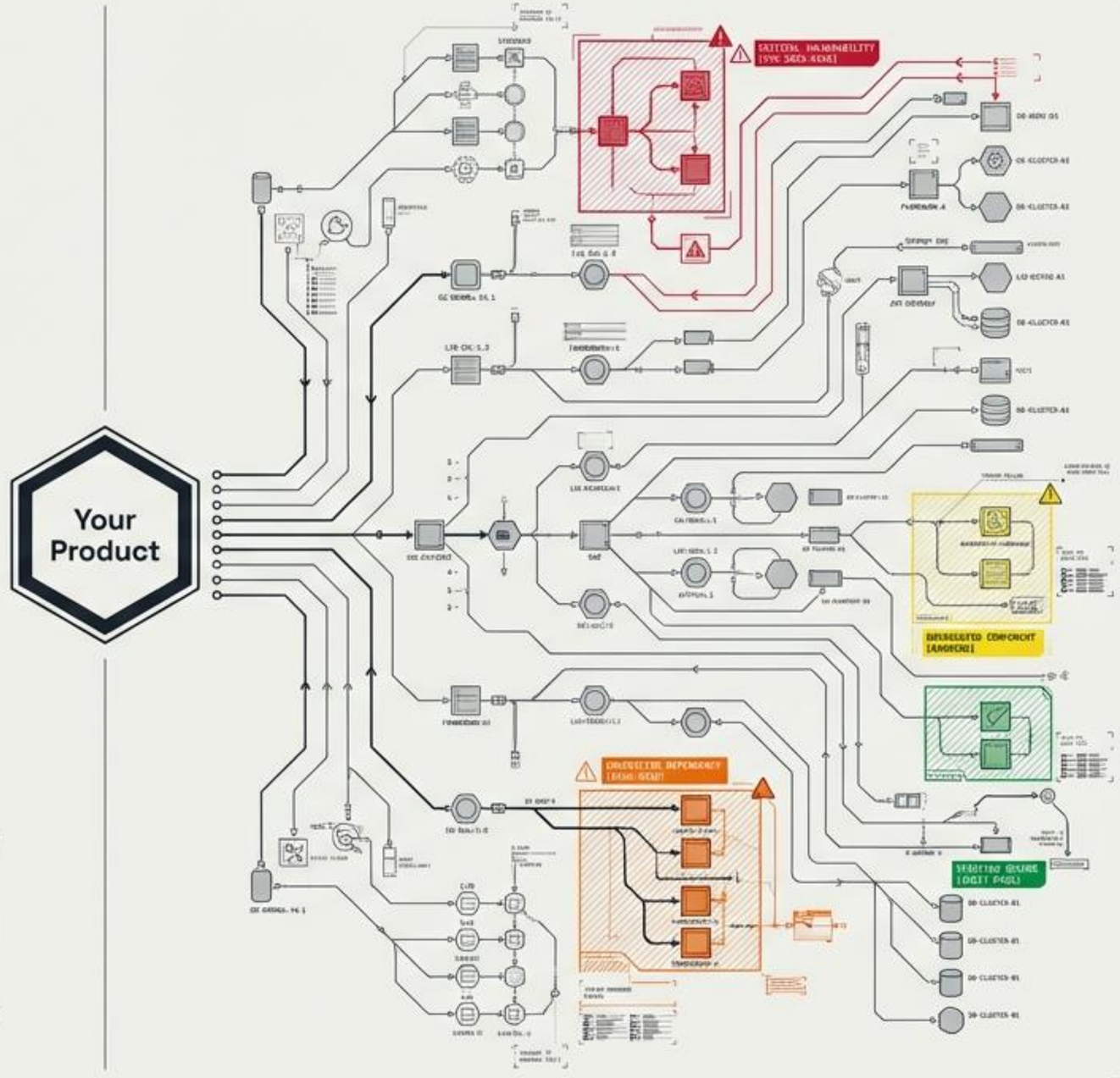


The problem isn’t the code you write; it’s the web of dependencies you inherit.

Modern products are not single binary files. They rely on external OS packages, frameworks, and protocols.

The Rule of Scale: The larger the component inventory, the greater the volume of unknown, unverified risk.

If we only look at the components, we drown in the noise.



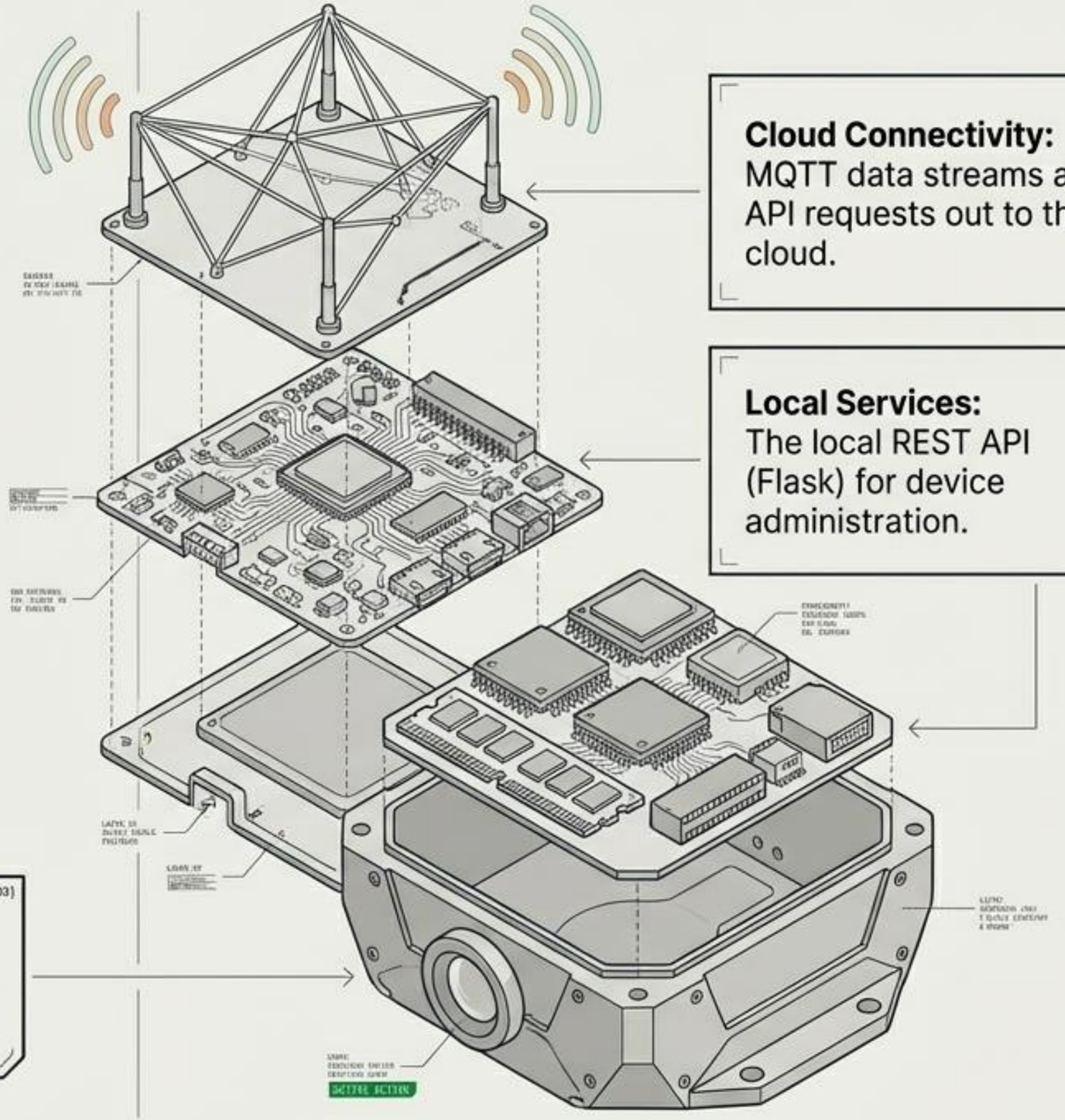
Meet the Stardust Meteor Dust Analyzer.

It’s not just a sensor. It’s an entire software supply chain trapped in a metal box.

Local Services:
The local REST API (Flask) for device administration.

OS / Firmware:
The base operating system (Debian) and privileged execution (sudo).

Hardware / Sensor:
The physical meteor dust acquisition layer.



Cloud Connectivity:
MQTT data streams and API requests out to the cloud.

Local Services:
The local REST API (Flask) for device administration.

The four operational questions defining CRA compliance.

The Compliance Funnel: Filtering noise into strategy.

SBOM (Inventory)

Answers: What do we have?

Provides the baseline transparency.

VEX (Context)

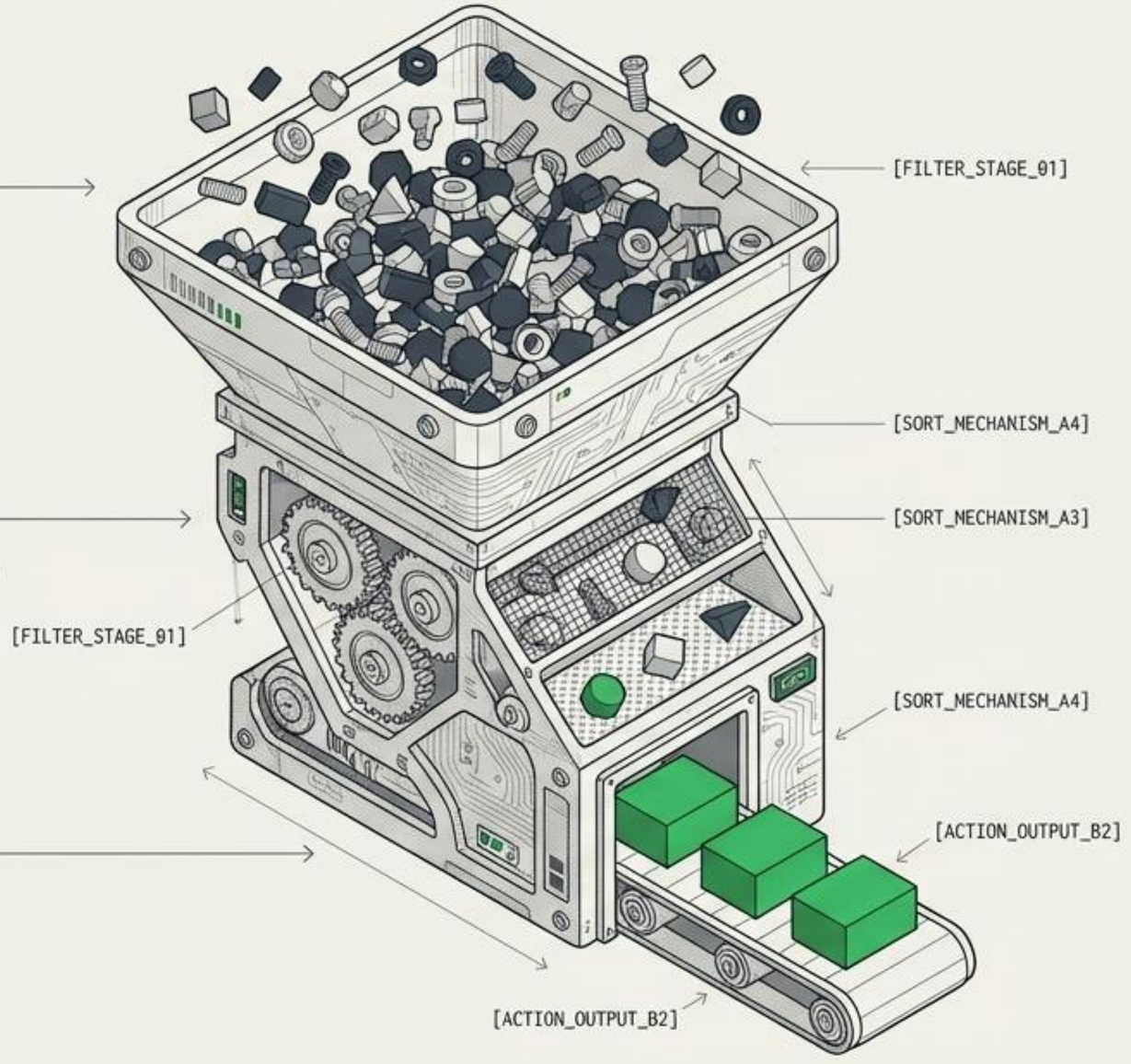
Answers: What actually matters?

Applies the deployment reality.

Risk Matrix (Action)

Answers: What do we do?

Outputs the business decision.



SBOM provides the absolute truth of your ingredients.

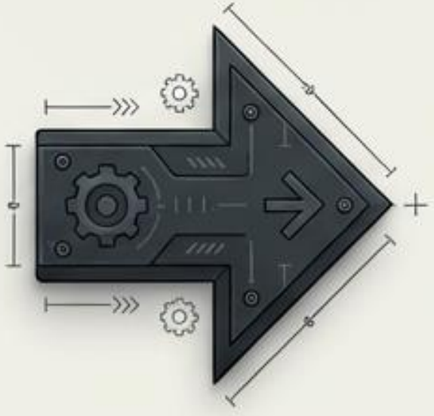
Without an SBOM, you are blind to what you ship. It creates a baseline for transparency, auditability, and vulnerability matching.

Nutrition Facts

PRODUCT SPECIFICATIONS

Ingredients:

- OS Kernel
- Communication Protocols
- Cryptographic Libraries
- Open Source Frameworks



```
{
  "bomFormat": "CycloneDX",
  "specVersion": "1.5",
  "metadata": {
    "component": {
      "name": "Stardust Meteor Dust Analyzer",
      "version": "0.2-demo"
    }
  }
}
```

Software Bill of Materials (CycloneDX v1.5)

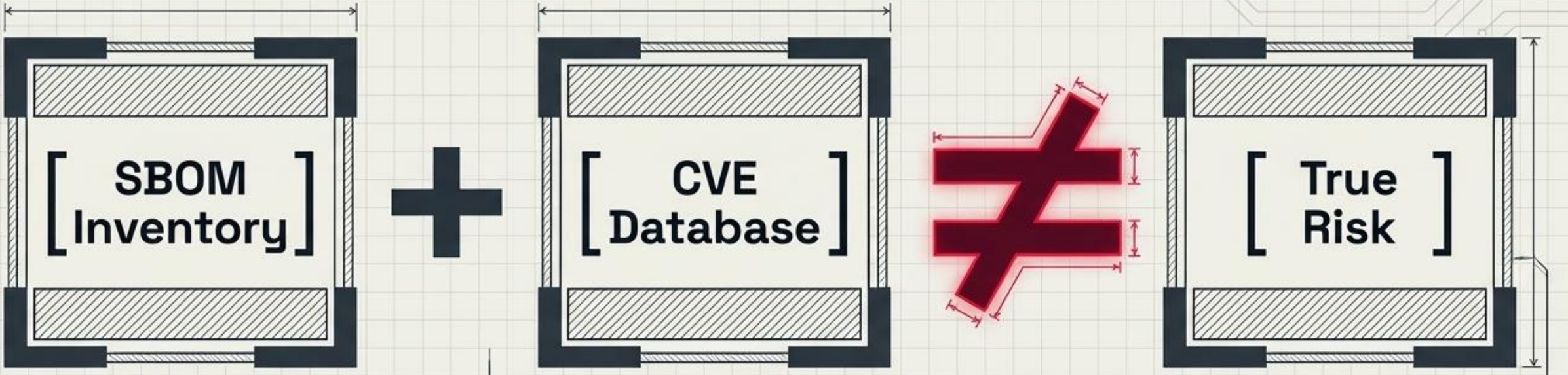
The Stardust component telemetry readout

Component Code	Version	Operational Role
os-debian-10	10	Base Operating System
sudo	1.8.27	Privileged maintenance
openssl	1.1.1j	TLS Crypto
flask	2.2.4	Local REST API
requests	2.31.0	Cloud API Client
mosquitto	2.0.15	MQTT Broker
stardust-sensor-agent	0.2.0	Local acquisition firmware

"bom-ref": "pkg:pyi/flask@2.2.4",

The illusion of vulnerability: Why SBOM data alone creates operational paralysis

Industrial Telemetry & Technical Blueprint



1

Cross-referencing flags everything blindly.

Creates noise, not actionable signal.

2

Cannot determine if vulnerability is actually exploitable.

Lacks runtime execution context.

3

ON

Blind to whether the component is active or dead code.

Resources wasted on inactive assets.

4

Ignores specific deployment context or network reachability.

Fails to account for environmental controls.

Blueprint Reference: VULN-ANALYSIS-202

The Scanner Reality: Every signal looks like a critical threat.

Under traditional compliance, engineering must drop everything to patch all five immediately. This is paralysis.

CVE-2021-3156 (sudo)
Critical - Privilege Escalation

CVE-2023-3592 (Mosquitto)
High - Memory Leak / DoS

CVE-2021-3449 (OpenSSL)
High - NULL pointer dereference

CVE-2024-35195 (requests)
High - Certificate bypass

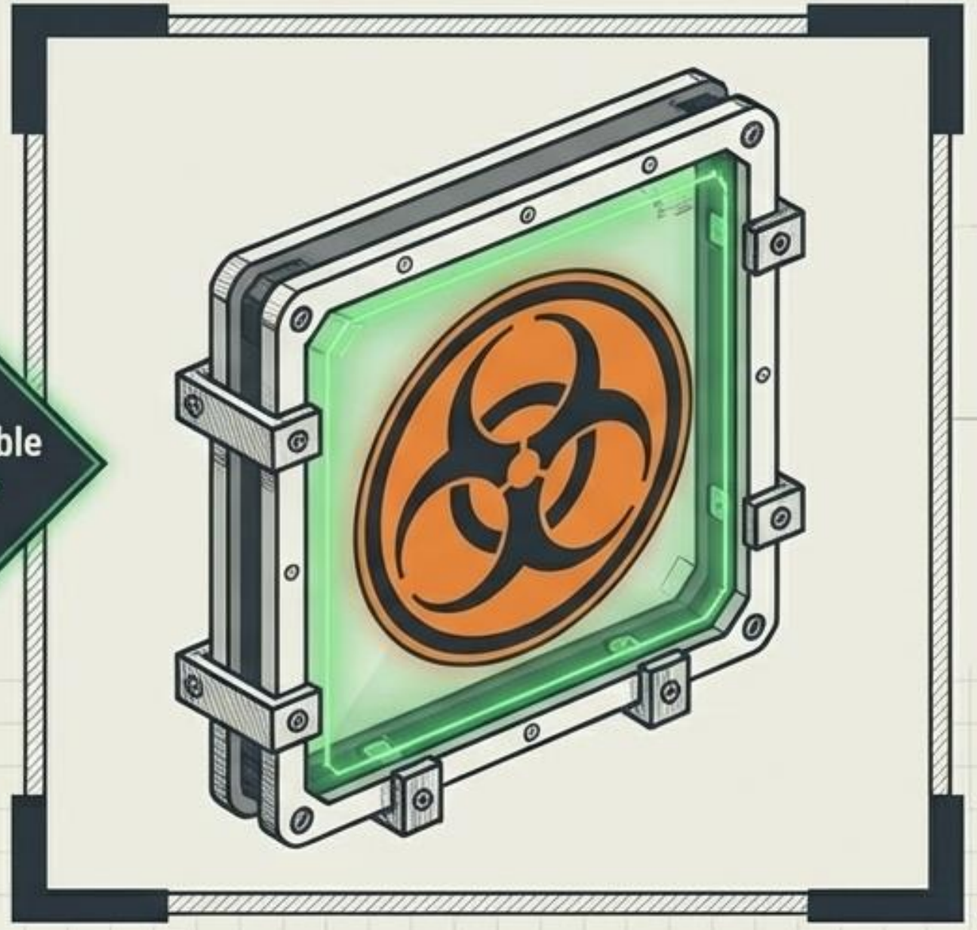
CVE-2023-30861 (Flask)
High - Session cookie disclosure

Blueprint Reference: VULN-SCANNER

VEX applies the missing dimension: Deployment Context.

Vulnerability Exploitability eXchange (VEX) transforms automated panic into an informed, defensible engineering decision.

Applies Machine-Readable Architecture Context →



The Ingredient Label

(What is inside?)

The Health Warning

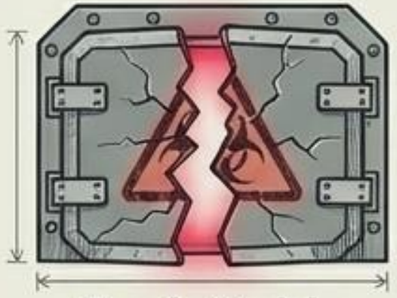
(Is it dangerous to consume right now?)

Blueprint Reference: VULN-ANALYSIS-005

The four definitive states of VEX evaluation.

These aren't just opinions; these are standardized, machine-readable cryptographic justifications.

CRITICAL



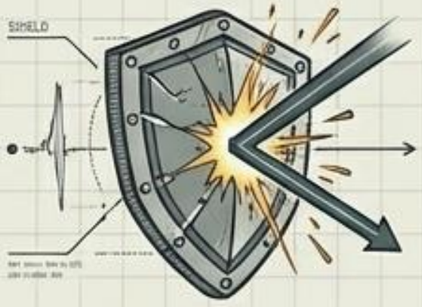
Exploitable

TRIAGE

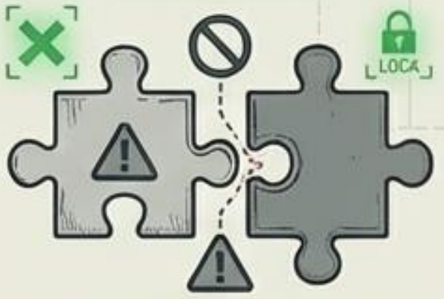
In Triage

MITIGATED

Mitigated



SECURE



Not Affected

Blueprint Reference: VEX-STATES-DEFINITION-001

Injecting Stardust's context into the VEX framework

Explicitly justifying scanner overrides through architectural reality.

Flask Evaluation

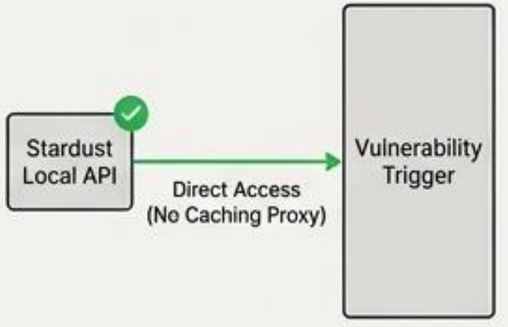
```
"bom-ref": "pkg:pypi/flask@2.2.4"
"state": "not_affected"
"justification": "inline_mitigations_already_exist"
```

300 pts

The Reality (Flask):

The Stardust local API is strictly direct-access. It does not sit behind a shared caching proxy.

The vulnerability is physically impossible to trigger.



150 pts

75 pts

requests Evaluation

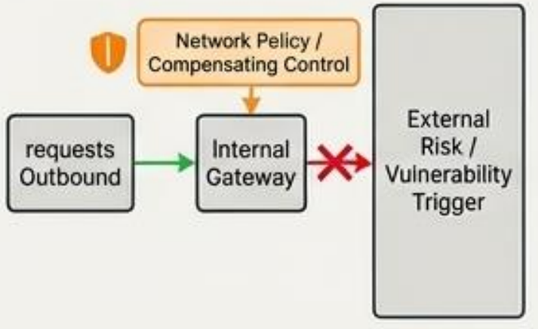
```
"bom-ref": "pkg:pypi/requests@2.31.0"
"state": "resolved"
"justification": "protected_by_mitigating_control"
```

300 pts

The Reality (requests):

Outbound calls are pinned to an internal gateway and blocked by network policy.

Compensating controls neutralize the risk.



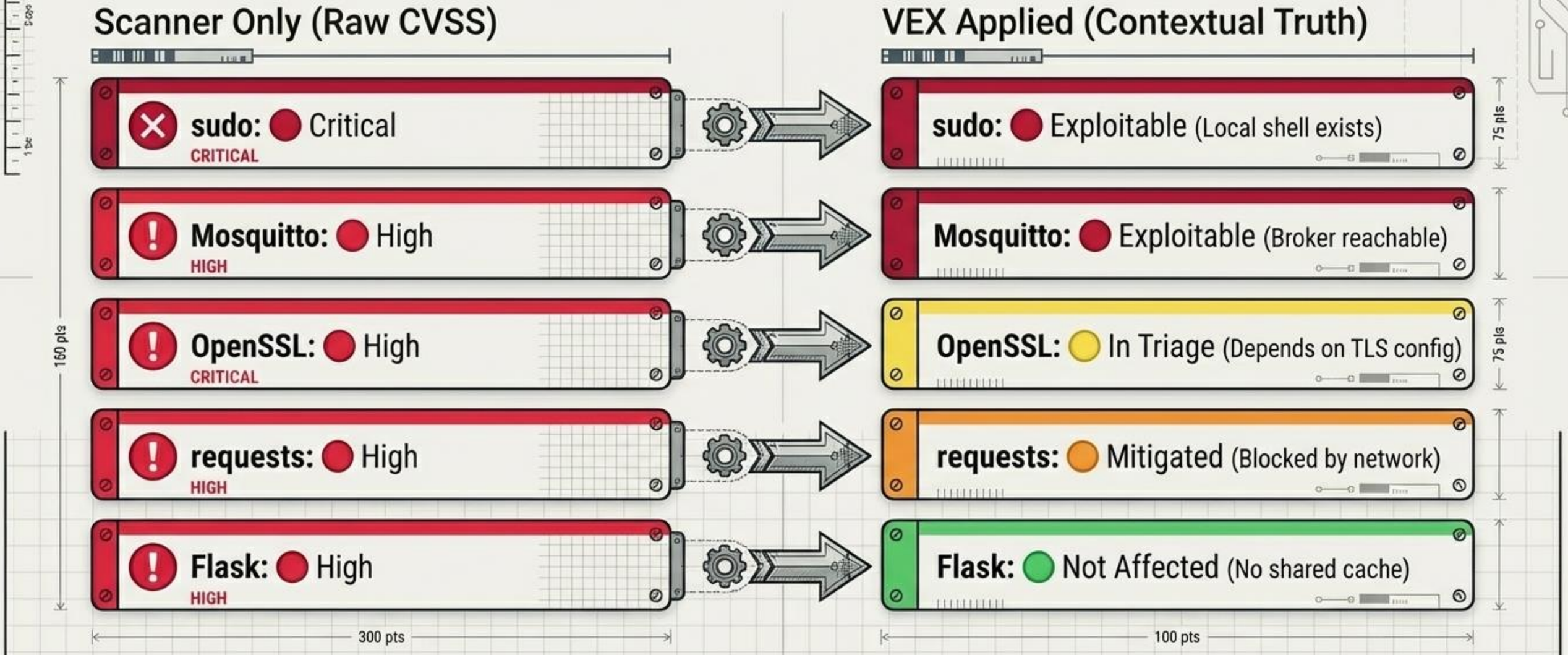
100 pts

75 pts

Blueprint Reference: STARDUST-CONTEXT-INJECTION-007

Diagnostic Comparison: Scanner Reality vs. VEX Reality.

Result: VEX instantly eliminated 40% of the critical workload through defensible context.



Blueprint Reference: VEX-STARDUST-COMPARISON-

Quantifying reality: The strategic risk equation.

Risk is not a static CVSS number. This equation creates an objective, mathematically sound defense for auditors.

MULTIPLIER SPECIFICATION TABLE

SPECIFICATION GRID

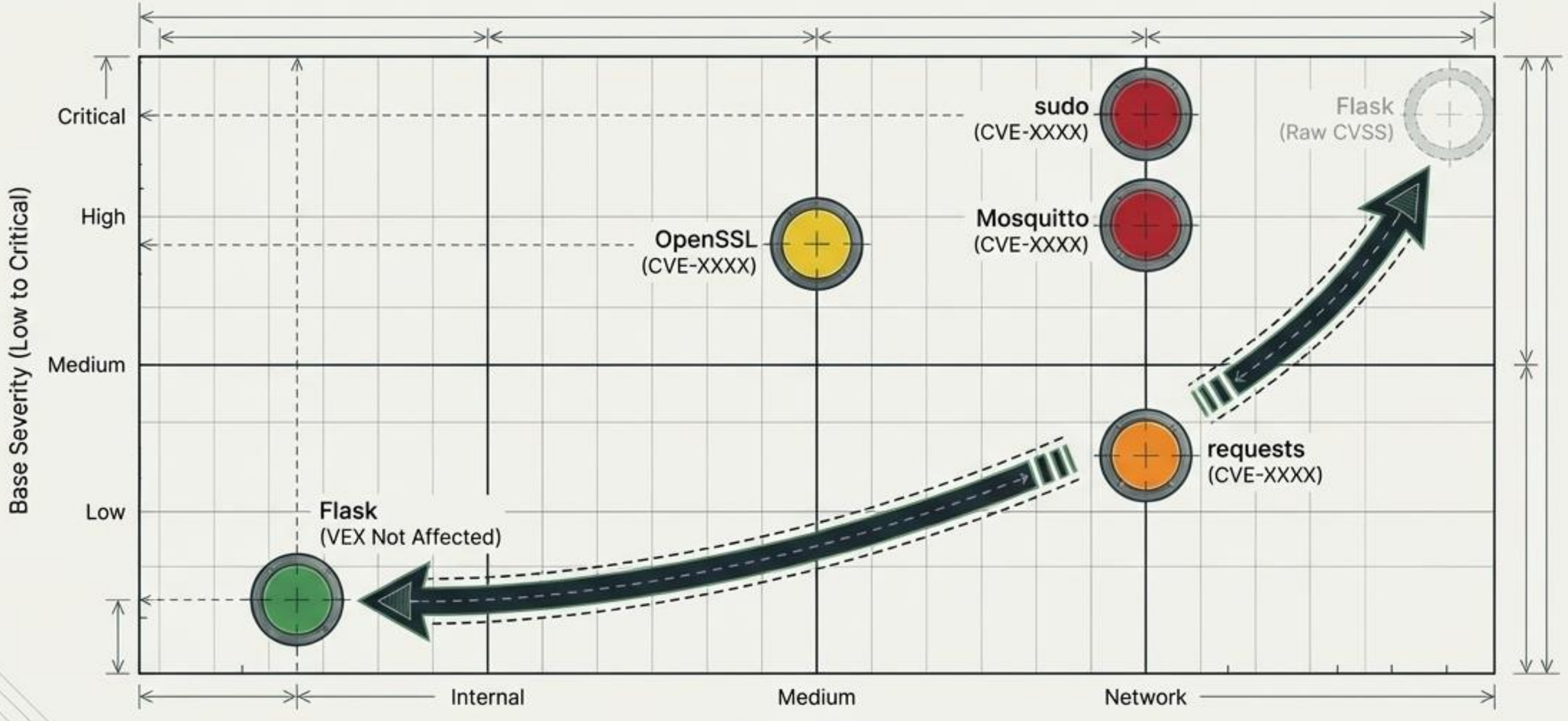
VEX STATUS	SCALAR MULTIPLIER	
Exploitable	x1.0	(Retains full risk)
In Triage	x0.7	(Discounted pending validation)
Mitigated	x0.4	(Compensating controls active)
Not Affected	x0.1	(Risk effectively zeroed)

75 pts
75 pts
300 pts

Blueprint Reference: RISK-EQUATION-SPEC-001

The Strategic Risk Assessment Matrix in action.

VEX physically moves the plot point of a vulnerability, adjusting its true threat vector based on operational controls.



RISK_MATRIX_V2.0

VEX_ADJUSTMENT_VECTOR_808

Translating VEX context into guaranteed Service Level Agreements.

SLA ACTION POLICY TABLE

SPECIFICATION: VEX-SLA-MAPPING-V3.1

Risk Level	Operational Action	Target SLA	Applies To
🔴 **High Risk (Exploitable)**	Immediate threat. Patch, isolate, or mitigate.	**7 days**	sudo, Mosquitto
🟡 **Medium Risk (In Triage)**	Potential risk. Investigate and validate config.	**14–30 days**	OpenSSL
🟠 **Low-Med Risk (Mitigated)**	Controlled risk. Upgrade in backlog.	**Next release sprint**	requests
🟢 **Low Risk (Not Affected)**	No current impact. Monitor only.	**N/A**	Flask

TARGET SLA (3L0 WIDTH)

SLA ACTION POLICY TABLE

SPECIFICATION: VEX-SLA-MAPPING-V3.1

Space Grotesk, Inter

Int-7

The Executive View: Defensible, prioritized intelligence.

The ROI: Engineering avoids wasting sprint capacity on phantom vulnerabilities, focusing only on legitimate vectors.

5

Total Raw CVE Signals

Initial scanner noise.

2

High Risk (Action Required Now)

Verified exploitable vectors actively compromising the device.

1

Medium Risk
(Investigate)

2

Low / Mitigated
(Deferred)

3 phantom vulnerabilities removed from immediate sprint.

EXECUTIVE_DASHBOARD_V3.1

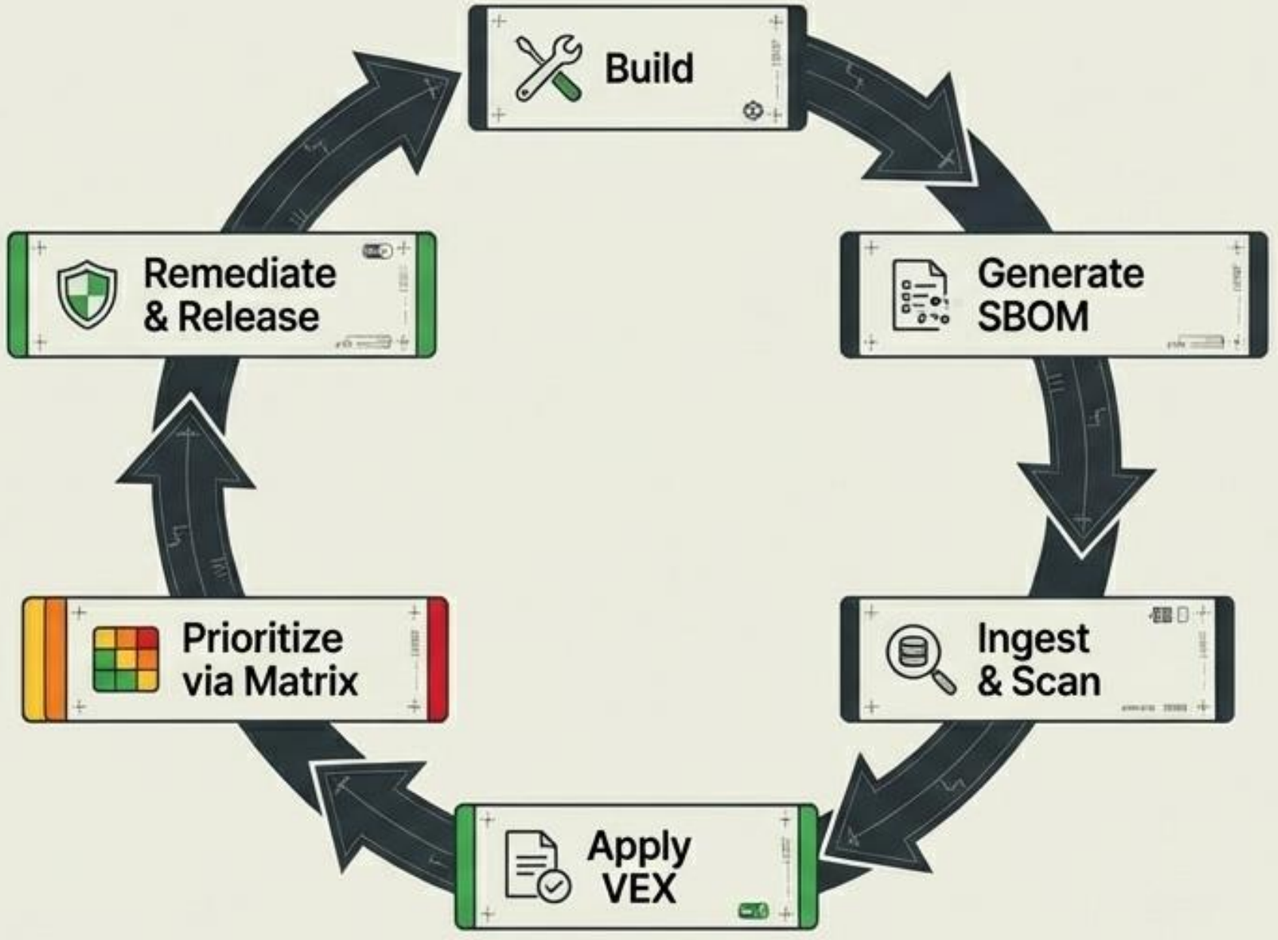
Space Grotesk, Inter

Int-8

Management, not a one-time document: The Continuous Lifecycle.

Compliance is **not not a static PDF**.
This cycle automatically triggers on:

- New software releases
- New dependencies
- Newly disclosed CVEs
- Audit or compliance reviews

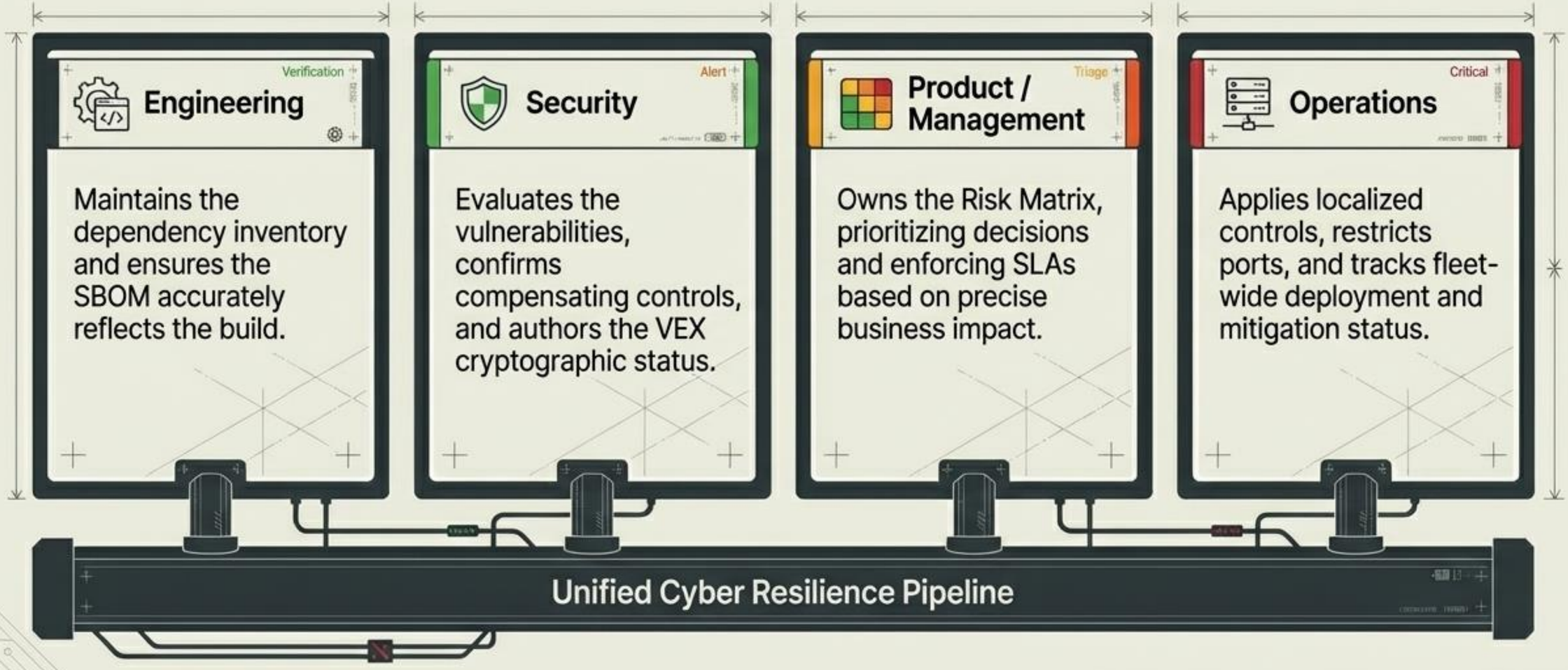


SPECIFICATION: CONTINUOUS-LIFECYCLE-V1.2

Space Grotesk, Inter | Int-9

Aligning organizational roles to the compliance engine.

Compliance is a distributed, cross-functional effort, not just a “security problem”.

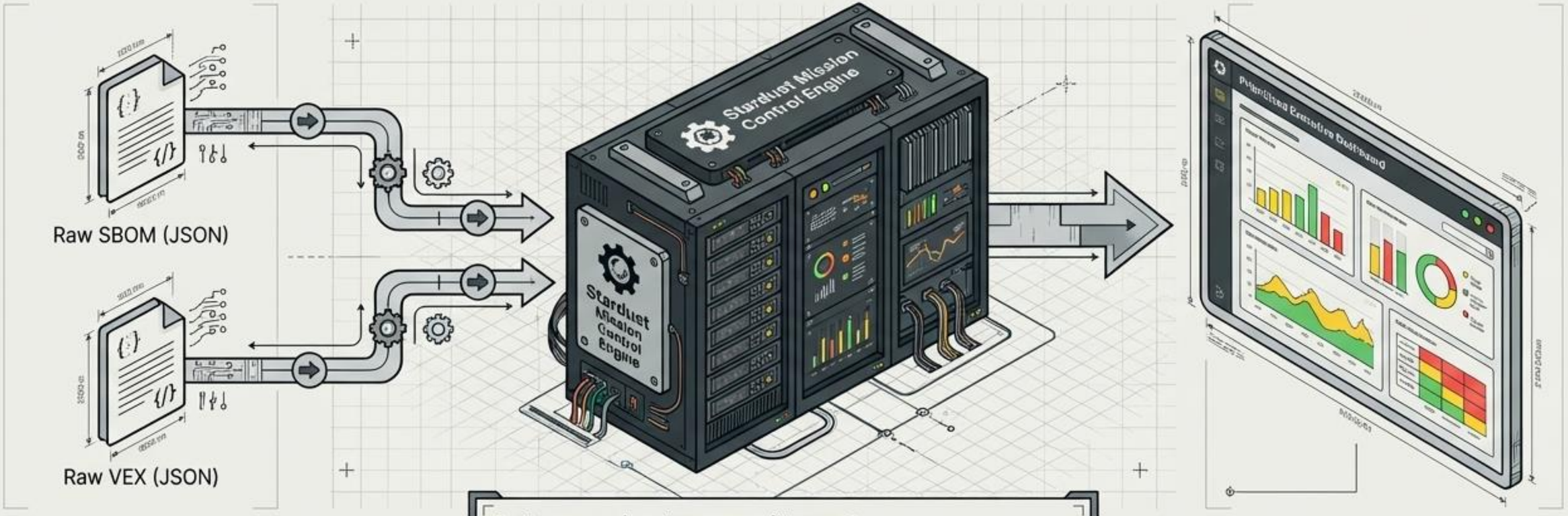


SPECIFICATION: ORGANIZATIONAL-ROLES-ALIGNMENT-V2.2

Space Grotesk, Inter | Int-10

Next step: From static documents to dynamic platform execution.

We understand the methodology. Now we automate the execution.



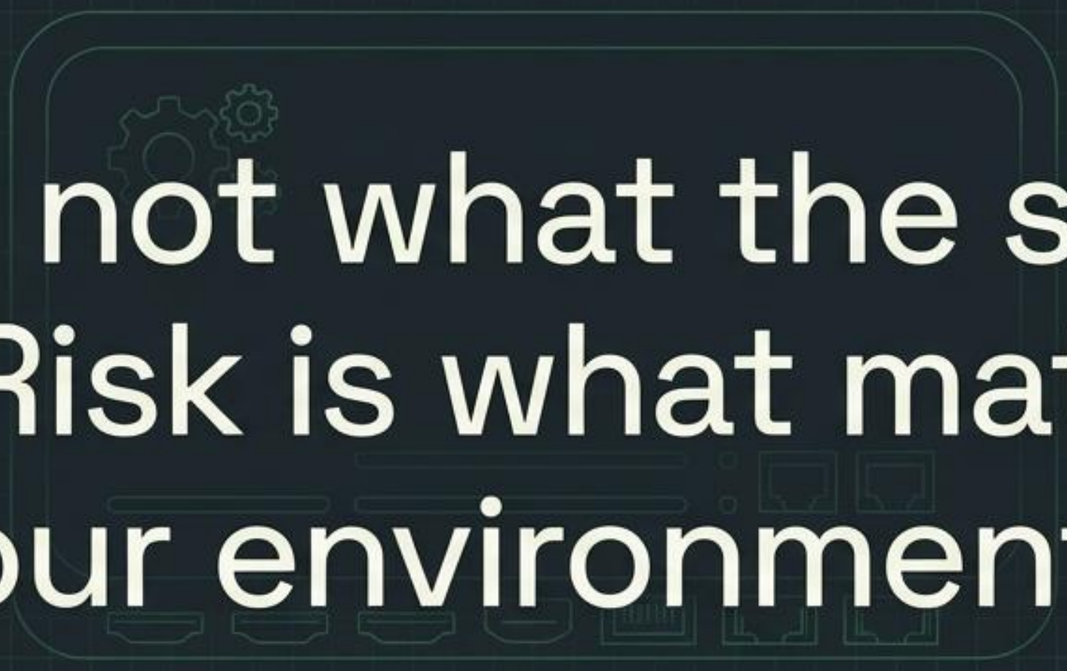
In the upcoming demo, we will execute:

- [✓] Automated ingestion of the Stardust CycloneDX SBOM.
- [✓] Real-time vulnerability mapping.
- [✓] Applying the VEX logic engine to override false positives.
- [✓] Generating the final, prioritized Risk Matrix dashboard.

SPECIFICATION: AUTOMATED-EXECUTION-PIPELINE-V2.1

Space Grotesk, Inter

Context is the only defense against chaos.



“Risk is not what the scanner finds. Risk is what matters in your environment.”