



# Linux Foundation OSS VulnDB Program - Overview

November 2025





# Background

# Background and Call to Action

**Situation** - Recent changes within the US government have brought several mission-critical vulnerability management & metadata programs into risk of halting operations (CVE, CWE, NVD) and no clear ecosystem leader has stepped forward to ensure the neutrality and continued availability of these global resources.

**Background** - In [February 2024](#), the National Vulnerability Database (NVD) began slowing down in processing and enrichment of vulnerability records. This was the first incident that highlighted the precarious nature of the vulnerability management ecosystem's dependence upon US government-funded programs. A second indication came April 2025, in a [leaked memo](#) that detailed that the CVE program administrator, MITRE, expecting to [cease operations](#) of the CVE and CWE programs, both of which are integral to how coordinated vulnerability disclosure operates today.

**Update** - Sept 10, 2025 CISA issues a two page high-level [document](#) describing their position and areas they state where effort will be investing.

The [current contract](#) is due to expire on **March 16, 2026**.

# Recommendation for Future

## **Recommendation:**

We propose expanding the existing Open Source Vulnerability database ([OSV](#)) ecosystem to extend the *already existing* global, federated vulnerability management system, anchored in open collaboration. This expanded system would build services, processes, and tooling to make it easy for projects, communities, and organizations to contribute data and participate actively across the federated network.

## **Potential path forward has been identified:**

Create a neutral, non-profit Foundation that would oversee this initiative, operating under the umbrella of the Open Source Security Foundation (OpenSSF). This new Foundation would drive cross-ecosystem coordination, stewardship of the federated model, and development of shared services (such as contribution tooling, discovery APIs, and shared trust and validation mechanisms) as well as hosting such infrastructure.

We will be assembling a broad coalition of SMEs from across industry and OSS foundations/projects to help develop and implement this program.

# 5 most important things OSS VulnDB can deliver to its stakeholders

## 1. **Analytics, Compliance, and Sustainability**

- a. Dashboards, exposure metrics, fix adoption tracking, and CRA/NIS2-aligned reporting for manufacturers and regulators. Backed by a sustainable, neutral foundation, ensuring the database won't collapse due to single-point funding or politics.

## 2. **Ecosystem Reach & Integration**

- a. A federated, neutral, and widely adopted system that synchronizes with existing sources (NVD, GHSA, OSV.dev, EUVD) and integrates into developer workflows (CI/CD, IDEs, SBOM tools). This makes it easier for projects, vendors, and users to participate and consume data.

## 3. **Upstream–Downstream Coordination & Transparency**

- a. Support for coordinated vulnerability disclosure (CVD), including embargo handling, maintainer validation, and CNA-like delegation. This reduces burden on open source maintainers while ensuring vendors and regulators get timely, accurate information.

## 4. **High-Quality, Machine-Readable Metadata**

- a. Rich, complete, and developer-friendly records (package name, version ranges, commits, SBOM links, severity, exploitability, fixes) in open formats (e.g., OSV Schema, SPDX, CycloneDX, VEX, CSAF). This ensures automation, better precision in vulnerability scanning, and fewer false positives for enterprise users.

## 5. **Trusted, Authoritative Vulnerability Identifiers**

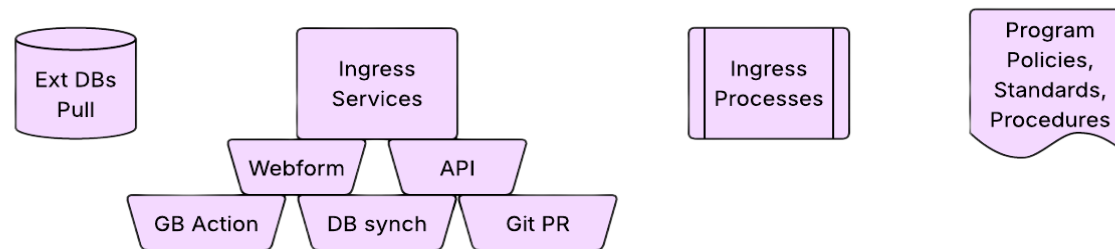
- a. A consistent, authoritative system of unique IDs for vulnerabilities (like CVE or OSV IDs) that supports global use, reduces confusion, and enables unambiguous communication across tools, researchers, vendors, and regulators.

# Overall System Design/Capabilities

All of the following slides are open for debate and are provided to help inspire further conversations about how we could implement such an upstream system

# Simple Design Diagram

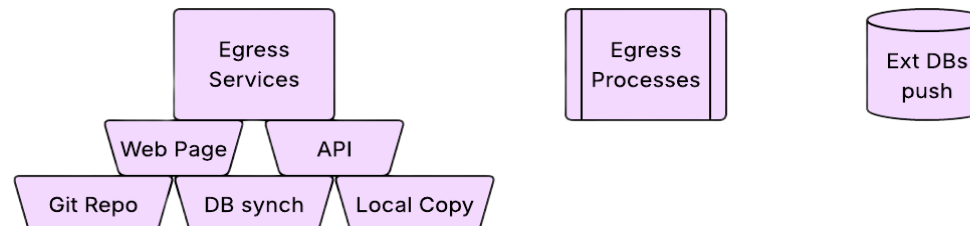
## Ingestion



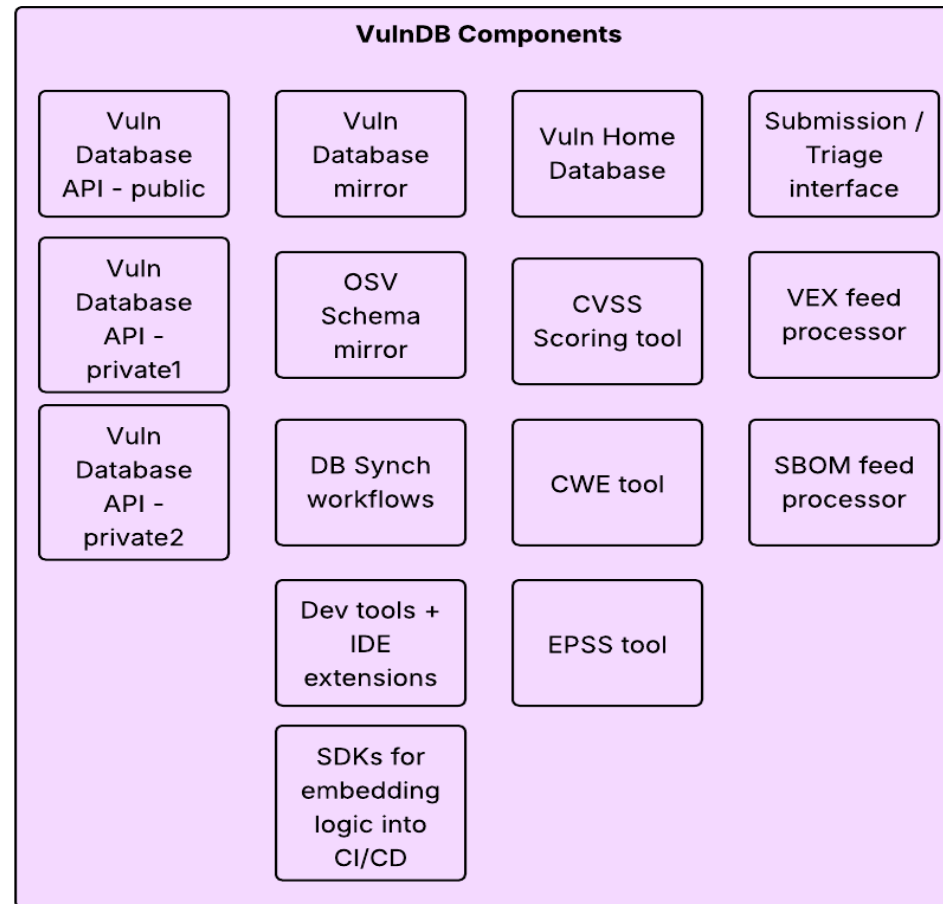
## Operations



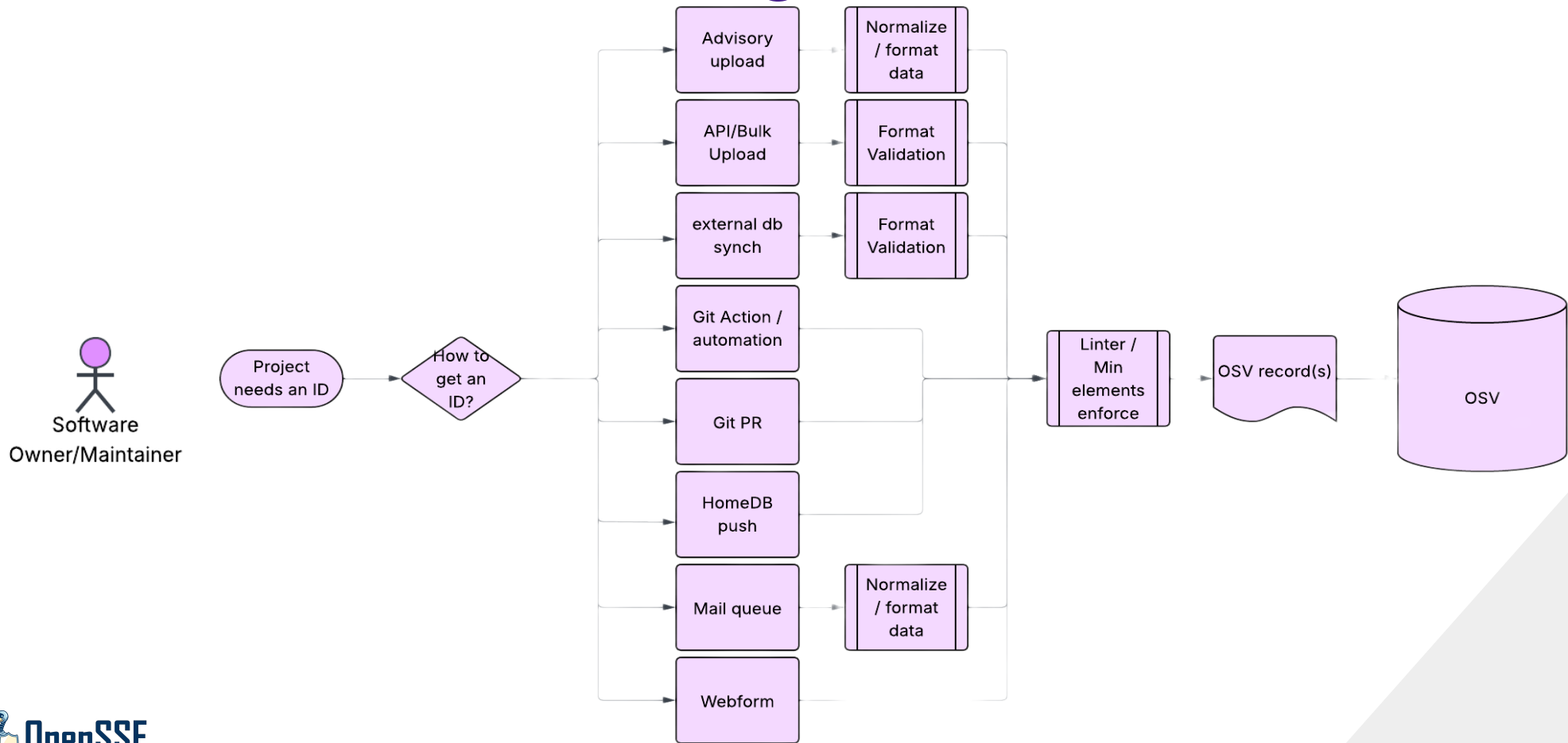
## Egress



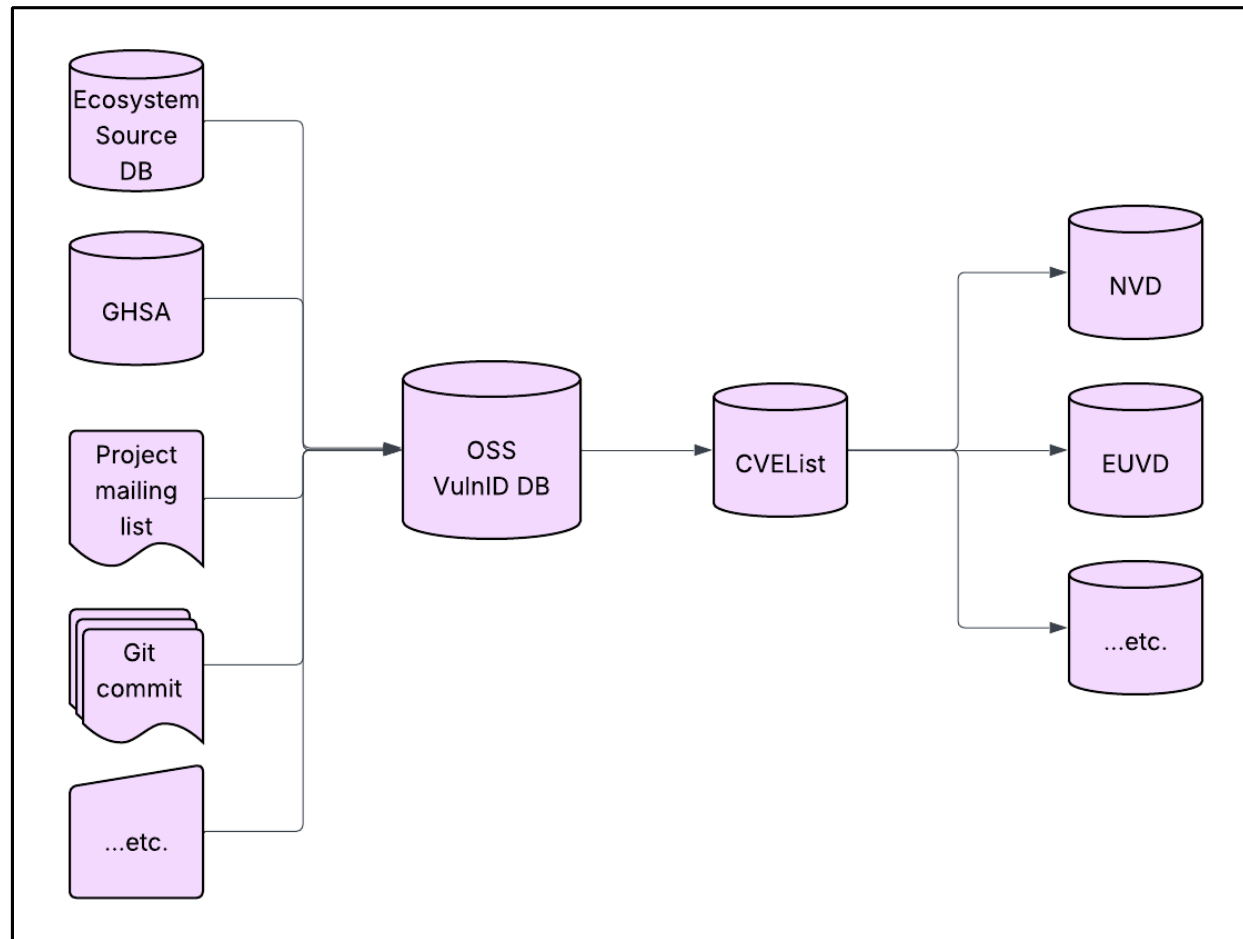
# Potential Services/Applications that support the system



# Potential Record Ingestion Workflow



# External DB interactions





# Program Benefits

# Benefits Summary

1. Built around the OpenSSF OSV schema and corresponding open, distributed vulnerability database hosted at [osv.dev](https://osv.dev)
  - a. This database aggregates vulnerability data from multiple open-source ecosystems and uses a simple **JSON schema** that precisely maps each vulnerability to package versions or commit hashes, making the data both human- and machine-readable
2. OSV's existing distributed home databases (operated by organizations such as GitHub, Canonical, Red Hat, Debian) **cover 28 ecosystems** (e.g., PyPI, npm and others) and the OSV Schema is widely adopted by major vendors
3. Tools exist within the OSV-ecosystem today that allow interoperability with the OSV format.
4. For PSIRT teams, this combination of breadth and automation means **faster identification of vulnerable components, more accurate dependency mapping**, and automated remediation.

# Why the OSV program is compelling for a corporate PSIRT

| Benefit to PSIRT                         | Evidence & rationale   |
|--|--|
| Comprehensive and timely data            | OSV aggregates vulnerabilities from many open-source projects and advisory feeds—including GitHub Security Advisories, PyPI, RustSec and others—using a unified schema. As of early 2025 it spans roughly 28 ecosystems. <i>This breadth gives PSIRTs a single source for most open-source vulnerabilities.</i>                          |
| Precise package/commit mapping           | The OSV schema includes fields that map vulnerabilities to specific version ranges or commit hashes. <i>This precision allows PSIRTs to match vulnerabilities directly to internal SBOMs, minimizing false positives and making it easier to prioritize remediation.</i>   |
| Machine-readable records and automation  | OSV records are in JSON and designed for machine consumption, enabling automated scanning and remediation (e.g., through <a href="#">osv-scanner</a> and CI/CD integration). <i>Machine-readable metadata also enables programmatic detection and analysis, which the CVE Programme has acknowledged is essential.</i>                   |
| Higher record quality                    | In contrast to many CVE records that lack usable <a href="#">.affected</a> fields or contain ambiguous version information, OSV's automated processes yield richer data, including version ranges and links to upstream patches. <i>This improves the quality of vulnerability intelligence and reduces triage time.</i>                 |
| Interoperability with SBOM/VEX workflows | OSV integrates with SBOM formats and tools and is being used in conjunction with VEX (Vulnerability Exploitability eXchange). Has alignment of identifiers (CPE, purl, etc.) <i>OSV's support for pURL identifiers and commit-level mapping helps PSIRTs correlate vulnerabilities with internal asset inventories more efficiently.</i> |
| Community-driven and resilient           | OSV is managed under the Open Source Security Foundation (OpenSSF) and receives contributions from Google, GitHub, Canonical, and many OSS communities. <i>Because it is open and distributed, it is less dependent on a single funder.</i>  |

## How the program can help a CNA today - Part 1

- Streamlined record creation and improved quality. Current CVE ID allocation is manual for many and slow, whereas **OSV uses automated processes to generate vulnerability records with commit-level precision**
- By using OSV as a source or reference when authoring CVE records, **CNAs can pre-populate fields** (affected version ranges, links to patches, etc.) **and avoid the manual guesswork that leads to invalid or missing .affected fields**
- **Better integration with modern identifiers (purl) and SBOMs**. Many CNAs still rely on CPE, which suffers from ambiguity and backlog issues
- OSV's support for package URLs (purl) and version ranges **enables CNAs to publish more actionable data** that can be ingested by modern vulnerability management tools.
- Greater efficiency and reduced operational cost. Because **OSV records are machine-readable** and available through an open API, CNAs can build automated pipelines to transform internal advisories into OSV format and cross-reference with existing CVEs
- This **reduces the manual overhead of maintaining a separate database** and improves ROI by allowing small teams to handle larger volumes of vulnerability data.

## How the program can help a CNA today - Part 2

- Access to a broader ecosystem and community. **OSV is part of a larger move toward a federated, community-driven model for vulnerability identification and dissemination**
- Leveraging OSV's infrastructure **allows CNAs to participate in this ecosystem, ensuring that their advisories reach downstream consumers quickly** and that they benefit from feedback and enhancements contributed by other stakeholders.
- Risk mitigation amid CVE instability.
- These initiatives highlight the risk of relying solely on CVE. By engaging with OSV and a federated OSS VulnDB program, CNAs help ensure continuity of vulnerability identification even if the CVE program experiences disruptions.
- Program would assemble a **broad coalition** of industry, community, and other interested parties to help make the processes and outputs **transparent, collaborative, and adaptable** to feedback and new requirements.

# Top 5 Problems with the current CVE Program

## 1. Delays in Assignment and Publication

- **Problem:** It can take **days to weeks** to get a CVE ID, especially if the reporter is not working with a well-established CNA (CVE Numbering Authority).

## 2. Uneven Coverage Across Ecosystems

- **Problem:** CVEs heavily favor **major vendors and operating systems**. Thousands of open source packages (especially npm, PyPI, Rust, etc.) go uncovered.

## 3. Rigid and Outdated Schema

- **Problem:** The CVE format doesn't handle modern development workflows well—like vulnerabilities tied to git commits, semantic versions, or SBOM integration.

## 4. Lack of Context and Quality Metadata

- **Problem:** Many CVEs lack sufficient detail—like which versions are affected, how to fix, links to advisories, or exploitability, and generally there are few paths to correct such inaccurate/incorrect data.

## 5. Gatekeeping and Bureaucracy

- **Problem:** Getting a CVE can feel political or inconsistent. Some researchers report rejections for “insufficient impact,” even when valid. Open Source developers complain about the lack of ease-of-use, automation, machine-readability

# User Stories and First Draft Requirements

[LF VulnDB User Stories](#)

[LF VulnDB Requirements - DRAFT](#)

# Full Implementation Benefits

## 1. Improved Timeliness and Automation

OpenSSF's VulnDB can issue identifiers and synchronize metadata rapidly through a modern, schema-driven infrastructure, eliminating current CVE bottlenecks in ID assignment and publication.

## 2. Comprehensive Ecosystem Coverage

By federating participation, the system will expand coverage to underserved areas like npm, PyPI, and Rust, correcting the ecosystem bias in today's vulnerability programs. System must be able to document both open source and commercial vendors; data.

## 3. Developer-Centric Architecture

Built on the OSV schema, the VulnDB supports git commits, semantic versioning, SBOMs, and machine-readable metadata--enabling seamless integration with CI/CD pipelines and automated tooling.

## 4. Global Trust and Governance

The Linux Foundation offers global, vendor-neutral governance--trusted by governments, commercial vendors, and open-source maintainers--critical for coordinating coordinated vulnerability disclosure (CVD) at scale.

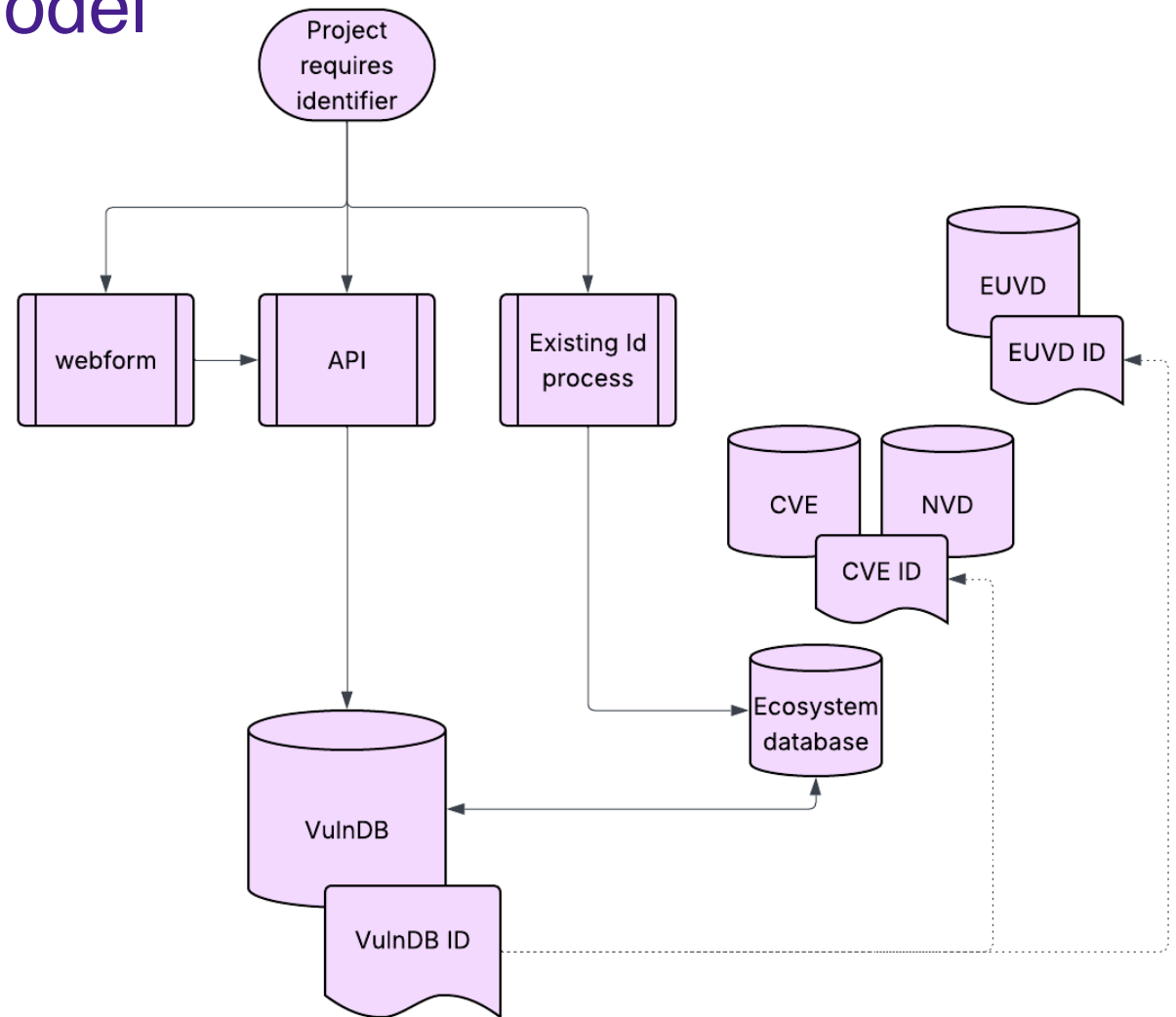
## 5. Compliance and CRA Readiness

By offering a unified, open data registry, the VulnDB supports compliance with emerging global regulations such as the EU Cyber Resilience Act (CRA), reducing burden for manufacturers and maintainers alike.

# ID Request Federated Model

One of the most crucial components of vulnerability disclosure is having a consistent way that all stakeholders use to refer to that discovered defect. This identifier is used throughout the disclosure process and throughout tooling that helps allow downstream to know when the vulnerability exists and later has been addressed (patched).

The proposed OVP program works along side and would federate with existing vuln identifier platforms.



# Benefits

- Stability for the industry
- Defines Open Source Security Foundation mission
- Unified upstream open source vulnerability processes and accessible data
- OSS-led program would reduce developer friction and frustration in the process; would provide researchers and Finders with an authoritative source of data and engagement
- Positions the program as THE go-to place for global governments to collaborate with on matters of cyber security and vulnerability disclosure in Open Source components.



# Thank you

