Compliance Detection Dashboard

O. Toure

October 2025

1 Project Introduction

The Compliance Detection Dashboard is being developed as a structured data system designed to identify relationships between regulatory policies and software vulnerabilities. The project integrates data from the Electronic Code of Federal Regulations (ECFR) and the MITRE Common Vulnerabilities and Exposures (CVE) database to establish a repeatable workflow for compliance review. The goal is to automate the collection, cleaning, and organization of both data sources into a structured format that supports deeper analysis and traceability.

This work reflects a practical application of data management within the context of compliance and cybersecurity. The dashboard is built to reinforce reliability, repeatability, and transparency in data processing. Each stage of the workflow, from ingestion to validation, is designed to be clear, testable, and directly applicable to real-world system assurance work. The output of this phase will serve as the foundation for future iterations that will extend into analysis and reporting.

2 Scope and Objectives

The project scope includes the design and implementation of a data ingestion and structuring pipeline capable of processing large unstructured datasets in both text and JSON formats. The primary objective is to create a framework where federal regulations from the ECFR can be analyzed alongside published vulnerabilities from MITRE's CVE data. By linking these two domains, the system supports an automated method for identifying where policy language and technical weaknesses intersect.

This iteration is focused exclusively on backend development. The emphasis is on building a stable ingestion and transformation pipeline that is reliable and well-documented. User-facing interfaces or alerting functions are intentionally out of scope for this phase. The objective is to establish a verified backend that future analytical tools can build on without reworking the foundational data flow.

3 Individual Contribution

This is an independent project managed and executed solely by the author. All development, testing, and documentation are handled directly, from setting up the development environment to writing scripts, defining schemas, and maintaining documentation.

The work draws on experience as a Cybersecurity Systems Engineer at Draper Laboratory, where compliance, traceability, and data integrity are integral to every phase of system design. That background informed the technical approach for this dashboard. The implementation applies skills in Python programming, API integration, and SQL database design to construct a repeatable process for managing compliance and vulnerability data. GitHub is used to version each update, while an Excel tracker maintains visibility across milestones and tasks.

4 Tools and Programming Environment

Development is done entirely in Python 3, using core libraries such as pandas, requests, BeautifulSoup, json, and sqlite3. SQLite serves as the database engine for structured storage and relational queries. GitHub manages version control and documentation synchronization. Overleaf is used for LaTeX reporting and final project submission.

The repository follows a structured layout: `/src` contains Python scripts, `/data` holds raw and cleaned datasets, `/reports` stores documentation and logs, and `/docs` includes references and supporting material.

This organization ensures that the entire process is repeatable and easy to extend. Every stage of development is documented and traceable to support academic evaluation and professional reproducibility standards.

5 Datasets and APIs

The project uses two publicly available APIs as primary data sources. The Electronic Code of Federal Regulations API (`https://www.ecfr.gov/developers/documentation/api/v1`) provides access to the full structure of federal regulatory text, including titles, parts, and sections. Each entry can be parsed into a relational schema for structured analysis.

The MITRE Common Vulnerabilities and Exposures API (`https://cveawg.mitre.org/api-docs/`) provides JSON data describing software and hardware vulnerabilities, including identifiers, descriptions, severity ratings, and affected product information. These datasets were selected because they are open, frequently updated, and represent critical intersections of compliance and risk management. Their integration demonstrates how regulatory and technical datasets can be unified within a single analytical

environment.

6 Project Alignment

The Compliance Detection Dashboard aligns with the objectives of DS5110: Introduction to Data Management and Processing. The project demonstrates proficiency in importing, cleaning, transforming, and structuring real-world data. It also applies disciplined version control, transparent documentation, and automation principles to ensure that all results are reproducible.

This work mirrors practices used in regulated environments at Draper Laboratory, where reliable data flow and auditability are essential. It extends classroom concepts into a realistic engineering workflow, bridging academic methods with professional compliance processes. The project also reflects how structured data systems can support cybersecurity assurance and compliance verification in large organizations.

7 Summary

The first phase of the Compliance Detection Dashboard establishes the technical and structural groundwork for the entire system. The data sources have been confirmed, the environment is configured, and the ingestion framework is in place. Each part of the workflow is documented and tested for reliability.

This report, along with the Excel progress tracker and GitHub repository, completes the deliverable package for Iteration 2. Together they represent the foundation of a scalable compliance data system that combines regulatory policy and vulnerability intelligence within a reproducible data architecture.