

Problem

Most cybersecurity tools weren't designed with small businesses in mind. The scanners exist, the frameworks exist, but all of it assumes you have a dedicated security team sitting behind it making sense of the output. Most small businesses don't have that. What they have is an owner wearing six hats, a Nessus export full of CVE codes, and no idea which one is about to cost them everything.

Solution

VulNerd is the nerdy classmate who actually wants you to pass a class you're behind in. It takes your scan, analyzes every finding, and hands you back a report card written like a tutor wrote it. Your biggest risks in plain English. What fixing them costs and what ignoring them costs. Students get a tool that bridges the gap between theory and real-world analysis. Small businesses get clarity without needing to hire a team to find it.

Tools

Tools used for this project include Python for data processing and server logic, DVWA as the vulnerable test environment, Remote Desktop Protocol (RDP) for remote access, a vulnerability scanner such as Nessus Essentials for data collection, Docker for environment deployment, Google Gemini AI for intelligent analysis and report generation, ReportLab for PDF report card generation, and a custom HTML, CSS, and JavaScript interface for visualization and delivery.

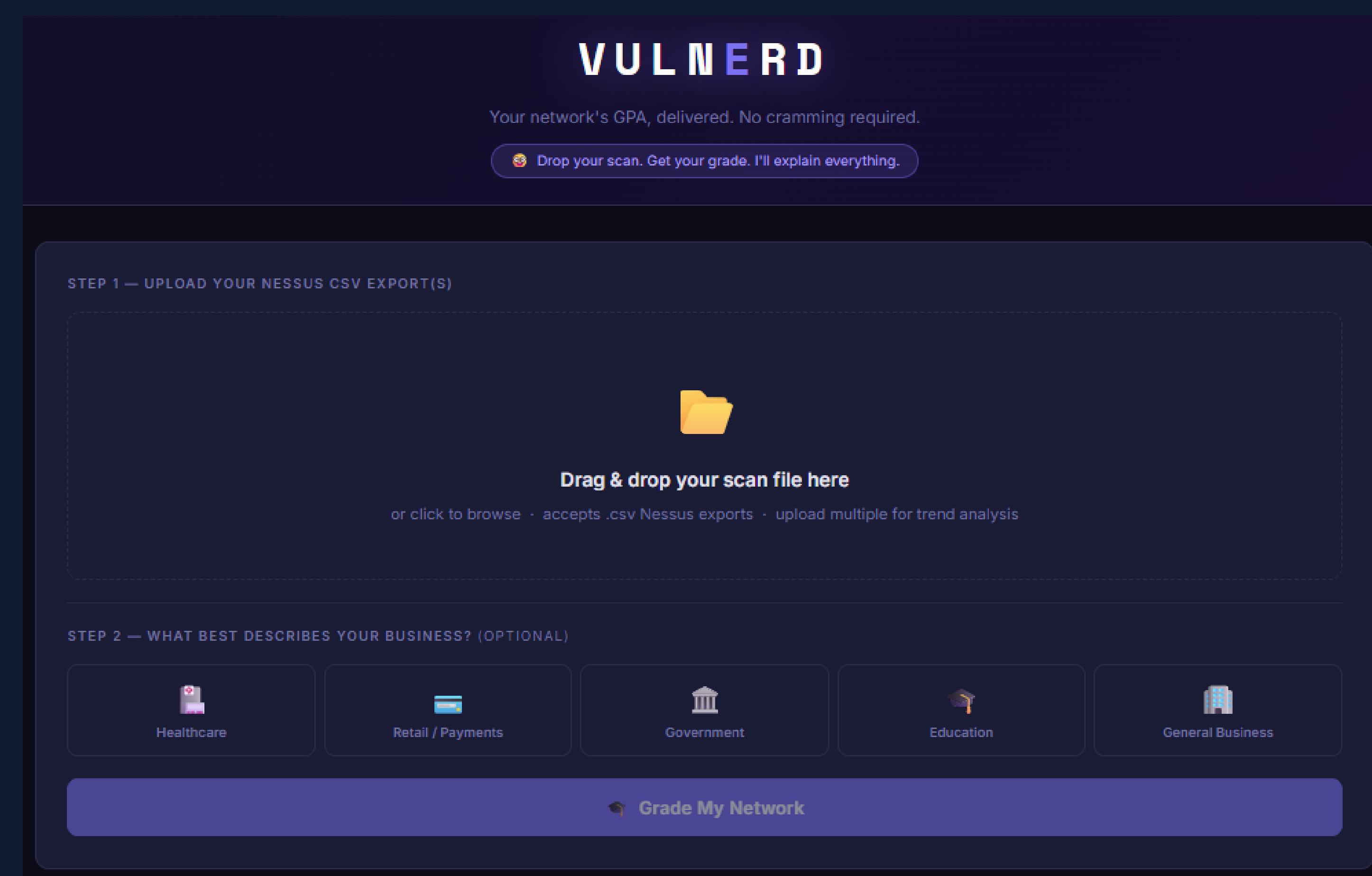


Students: Reynaldo Rodriguez, Jonathan Nava-Arenas, Luisa Faria Novy E Silva, Anthony Whiteman, Rickoy Cunningham

System Design

The VulNerd system is a modular vulnerability analysis pipeline that transforms raw scan data into structured, actionable insights.

- Data Collection: A vulnerability scanner such as Nessus Essentials performs scans on a DVWA environment
- Data Processing: Scan results are exported as CSV and parsed using a custom Python engine
- Data Enrichment: CVSS scores, severity levels, affected services, and vulnerability details are extracted and normalized
- AI Analysis: Google Gemini reads the enriched findings and generates a plain English report card with grades, risk explanations, and remediation guidance
- Data Visualization: Results are displayed through a custom web interface with a compliance dashboard, PDF export, and built in AI chat assistant
- Analysis Workflow: Users prioritize vulnerabilities, track improvements after mitigation, and measure security progress over time



Implementation

- Deployed DVWA in a controlled Docker based lab environment
- Conducted vulnerability scans using a vulnerability scanner such as Nessus Essentials
- Exported scan results and built a Python based parsing engine to clean, normalize, and structure scan data
- Integrated Google Gemini AI to generate graded, plain English report cards with risk explanations and remediation guidance
- Developed a custom web interface with a compliance dashboard, PDF export, and built in AI chat assistant for interactive analysis
- Implemented CVSS based prioritization for realistic risk assessment
- Performed iterative scans to validate mitigation effectiveness

SCAN: TARGET_SCAN_B131ZV.CSV		
SEVERITY	COUNT	DEDUCTION
Critical	22	-440 pts
High	0	-0 pts
Medium	33	-99 pts
Low	12	-12 pts
Total Findings	67	0/100

"This report requires immediate escalation. The findings represent serious exposure and should be reviewed with your security team today."

Team Time Line



1. Provisioned a DVWA-based lab environment with logging and visibility instrumentation.
2. Performed controlled adversary emulation to generate security telemetry.
3. Developed and refined detection logic to minimize false positives.
4. Correlated findings with external threat intelligence for contextual risk analysis.
5. Developed a CSV-driven analytical interface with visualizations and automated reporting.