Setup & Usage Guide

# 1. Overview

Vulnerability Module is a Python-based analysis pipeline that detects known vulnerabilities in network captures or scan results. It supports PCAPs and Nmap XML files, and maps observed activity to CVEs using a merged dataset from ExploitDB and NVD.

# 2. Installation

Requirements:  
- Python 3.8+  
- tshark (for PCAP extraction)  
- pip packages: pandas, pyVIP  
- Virtualitics Explore (optional, for 3D visualization)

Setup Steps:  
1. Clone or download this repository  
2. Install dependencies:  
 pip install -r requirements.txt  
3. Install tshark:  
 https://www.wireshark.org/download.html  
 Ensure it's in your system PATH  
4. (Optional) Install pyVIP:  
 pip install pyVIP  
5. Set your VIP\_AUTH\_TOKEN as an environment variable or enter it manually when prompted

# 3. Folder Structure

Vulnerability/  
├── main.py # Main runner script  
├── core/ # Extraction + matching logic  
├── data/ # Inputs and analysis results  
│ ├── input/ # Raw input files (.pcap/.xml)  
│ └── output/ # Extracted and matched output files  
├── exploit\_db/ # NVD + ExploitDB sources and merged output  
│ ├── nvd/ # NVD JSON and flattened CSVs  
│ ├── exploitdb/ # Raw ExploitDB data  
│ ├── index/ # Merged + indexed datasets  
│ └── custom/ # (Optional) Proprietary signatures  
├── tools/ # pyVIP and optional utilities  
├── docs/ # Setup guides and internal docs  
├── requirements.txt # Python dependencies  
└── README.md # Project overview

# 4. Running the Pipeline

1. Drop a `.pcap` or `.xml` file in `data/input/`  
2. Run: python main.py  
3. Enter the input file path or press Enter to use the default

Output will be saved to `data/output/`:  
- pcap\_matches.csv (for PCAP)  
- nmap\_matches.csv (for XML)

# 5. Visualizing with Virtualitics

Run: python tools/pyvip\_exploit\_scatter.py

You will be prompted to enter:  
- Your pyVIP API token  
- Path to match results (CSV)

Virtualitics Explore will launch a 3D scatter plot for interactive review.

# 6. Updating the Vulnerability Database

Run the following to refresh CVE data:  
python core/nvd\_flattener.py  
python core/exploitdb\_merger.py  
python core/exploitdb\_indexer.py

These scripts rebuild the merged CVE + ExploitDB index.

# 7. Optional Tools

- pyVIP-based visualization via tools/pyvip\_exploit\_scatter.py  
- Expandable tool space for future reports, dashboards, etc.

# 8. Tips & Troubleshooting

Problem | Solution  
------------------------------- | ----------------------------------  
tshark not found | Add to PATH or reinstall Wireshark  
VIP\_AUTH\_TOKEN error | Export token or input manually  
No matches found | Ensure inputs reference known CVEs  
pyVIP not launching Explore | Confirm Explore is running and signed in