User Manual:

AI Based Network Based Intrusion Detection System- Implementation

1. Introduction

This document outlines the complete process to install, configure, and operate our **Project- AI-Based Network Intrusion Detection System**. This system represents a hybrid approach to cybersecurity: utilizing **Random Forest Classifiers** for real-time packet filtering (98%+ accuracy) and **Generative AI (Groq Llama 3)** for automated forensic analysis.

Core Capabilities:

- **Hybrid Data Engine:** Train on internal mathematical simulations or load external industrial datasets (CIC-IDS2017).
- **Live Attack Simulator:** A "Red Team" module to manually inject specific packet parameters (DDoS patterns) to test defense mechanisms.
- **AI Analyst Dashboard:** A specialized console that provides human-readable explanations for why traffic was flagged as benign or malicious.
- **Forensic Auditing:** One-click generation of professional PDF Security Reports.

---

2. Prerequisites

Ensure your environment meets the following requirements:

- **Operating System:** Windows 10/11, macOS, or Linux.
- **Python:** Version 3.10 or higher.
- **API Key:** A free API key from Groq Cloud Console (Required for the AI features).
- **(Optional) Dataset:** If testing real data, download *Friday-WorkingHours-Afternoon-PortScan.pcap_ISCX.csv* from the CIC-IDS2017 Dataset.

---

3. Installation

Step 1: Download Source Code

1. Clone or download the project repository to your local machine.

```
git clone https://https://github.com/hi-ashup/AI-Based-Network-Intrusion-Detection.git
```

2. Navigate into the project folder. `cd VOIS-NIDS`

Step 2: Install Dependencies

- The system relies on specific scientific computing and GUI libraries.
- Execute the following command to install them automatically:

- Open your terminal in the project directory and run: `pip install -r requirements.txt`

(This installs Pandas, Scikit-Learn, Matplotlib, Seaborn, and Requests).

Step 3: Launch the Dashboard

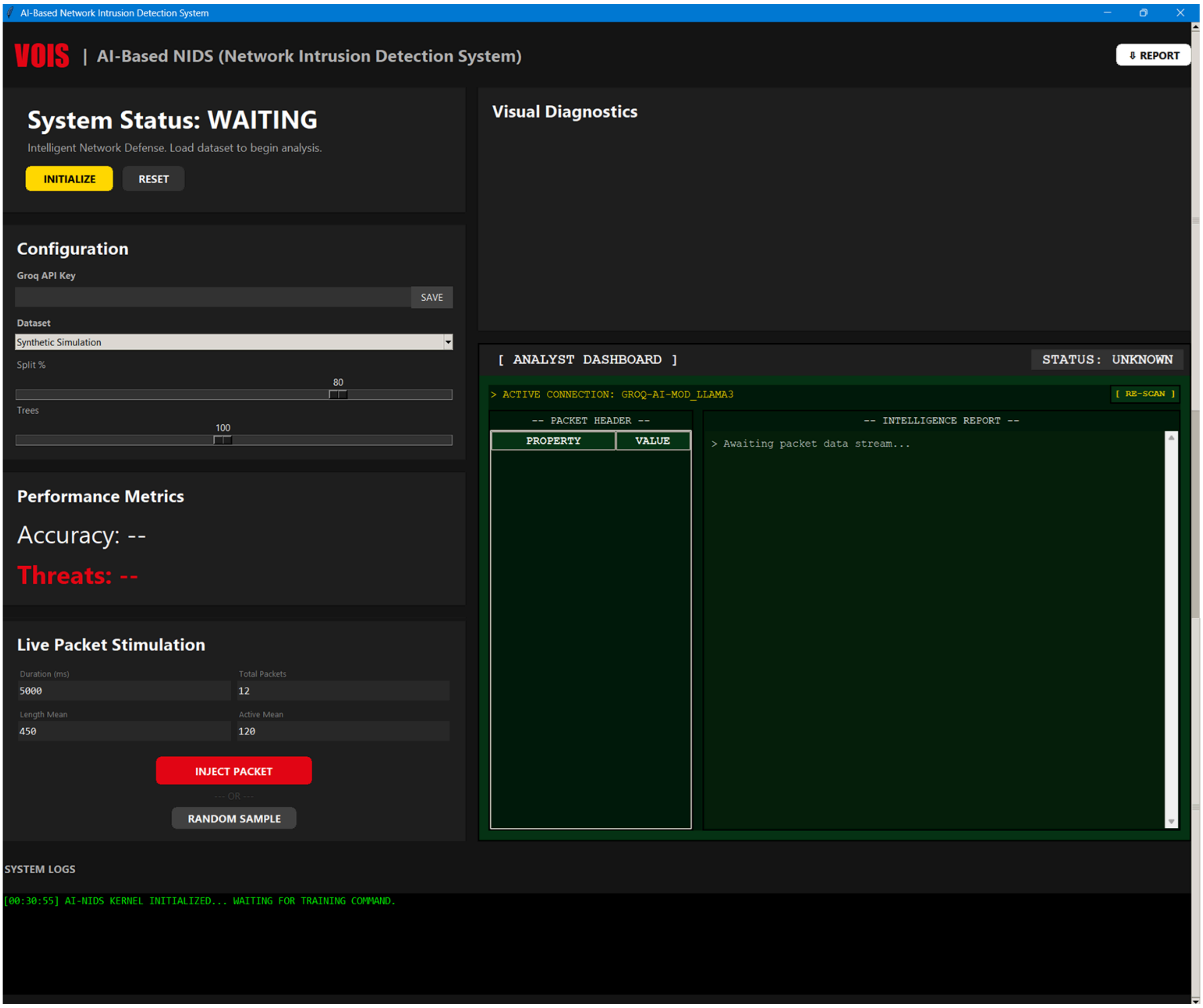
Run the main application script: `python src/main.py`

A dark-themed high-contrast window titled "AI-Based Network Intrusion Detection System" should appear.

---

4. Dashboard Overview

The interface is divided into a high-contrast layout optimized for readability:



AI-Based Network Intrusion Detection System

VOIS | AI-Based NIDS (Network Intrusion Detection System)

System Status: WAITING

Intelligent Network Defense. Load dataset to begin analysis.

Configuration

Groq API Key

Dataset: Synthetic Simulation

Split %: 80

Trees: 100

Performance Metrics

Accuracy: --

Threats: --

Live Packet Stimulation

Duration (ms): 5000

Total Packets: 12

Length Mean: 450

Active Mean: 120

Visual Diagnostics

[ANALYST DASHBOARD]

STATUS: UNKNOWN

```
> ACTIVE CONNECTION: GROQ-AI-MOD_LLAMA3
```

-- PACKET HEADER --

PROPERTY	VALUE

-- INTELLIGENCE REPORT --

```
> Awaiting packet data stream...
```

SYSTEM LOGS

```
[00:30:55] AI-NIDS KERNEL INITIALIZED... WAITING FOR TRAINING COMMAND.
```

LEFT PANEL (Control & Input)

1. **Configuration:** Settings for API Keys, Dataset Source, and AI Hyperparameters (Split % / Trees).
2. **Performance Metrics:** Displays the Model Accuracy and total Threats Intercepted once trained.
3. **Live Packet Stimulation:** Inputs for manually defining packet attributes (Duration, Size, Frequency) to simulate traffic.

RIGHT PANEL (Analysis & Output)

1. **Visual Diagnostics:** A Confusion Matrix heat map showing true positives/negatives.
2. **Analyst Dashboard:** A stylized console displaying raw packet headers alongside the syntax-highlighted AI Report.

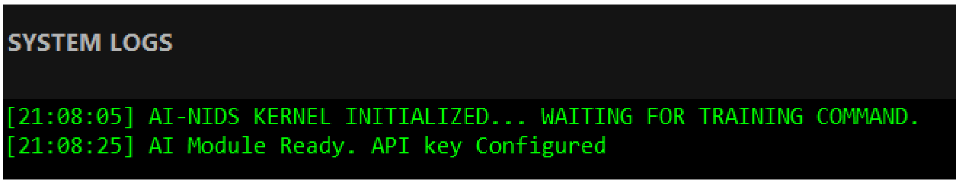
---

5. Operational Walkthrough

Phase 1: System Configuration & Training

Upon launch, the system status is "**WAITING**". The engine must be trained before it can detect threats.

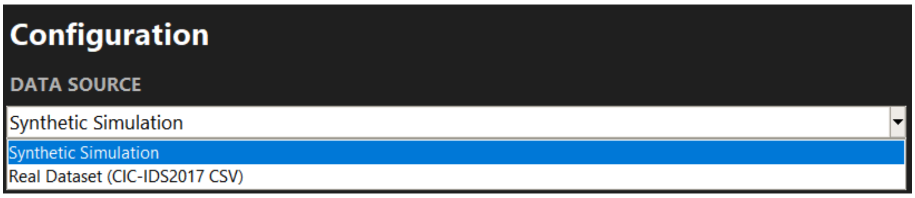
1. **Connect AI Brain:**
   - Locate the **Configuration** panel (Top Left).
   - Paste your key starting with `gsk_...` into the **Groq API Key** field.
   - Click **SAVE**.
   - *Verification:* Check the "System Logs" terminal at the bottom. It should read: `AI Module Ready`.



SYSTEM LOGS

```
[21:08:05] AI-NIDS KERNEL INITIALIZED... WAITING FOR TRAINING COMMAND.
[21:08:25] AI Module Ready. API key Configured
```

2. **Select Data Engine:**
   - **Mode A: Synthetic Simulation (Recommended for Demos)**
     - Select `Synthetic Simulation` from the dropdown. The system will mathematically generate 5,000+ normal and attack records internally.

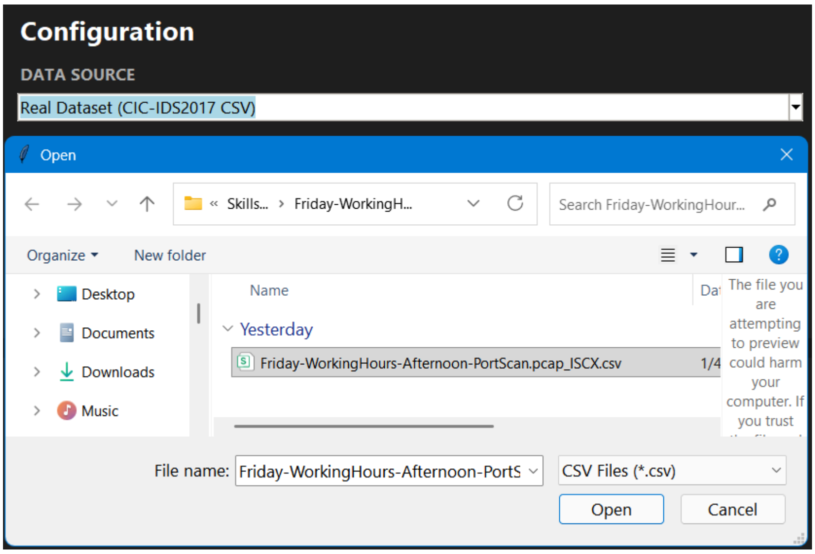


Configuration

DATA SOURCE

Synthetic Simulation

- **Mode B: Real-World Dataset (Advanced Forensics)**
    - Select Real Dataset (CIC-IDS2017 CSV).
    - A file dialog will open. Select your Friday-WorkingHours...pcap_ISCX.csv file.
    - *Note*: The training process may take 10-30 seconds depending on file size.



Configuration

DATA SOURCE

Real Dataset (CIC-IDS2017 CSV)

File name: Friday-WorkingHours-Afternoon-PortS

CSV Files (*.csv)

Open
« Skills... › Friday-WorkingH...
Search Friday-WorkingHour...
Organize
New folder
Desktop
Documents
Downloads
Music
Name
Da
Yesterday
Friday-WorkingHours-Afternoon-PortScan.pcap_ISCX.csv
1/4
The file you are attempting to preview could harm your computer. If you trust
Open
Cancel

3. **Train the Model:**
    - Adjust **Split %** (Default 80%) and **Trees** (Default 100) if desired.
    - Click the Yellow **INITIALIZE** button.
    - **Success Indicator:** The **Performance Metrics** panel will update (e.g., *Accuracy*: 98.45%), and the **Visual Diagnostics** chart on the right will render a Confusion Matrix.

Phase 2: Live Packet Stimulation (Red Teaming)

Once initialized, you can test the system's defenses by simulating network traffic.

Scenario A: Benign Web Traffic

Simulate a normal user slowly browsing a website.

1. Go to the **Live Packet Stimulation** panel (Bottom Left).
2. **Input Parameters:**
    - Duration: 5000 (ms) -> *Long connection time*
    - Total Packets: 5 -> *Minimal data exchange*
    - Length Mean: 450
    - Active Mean: 50
3. Click **INJECT PACKET**.
4. **Result:**
    - Status Banner turns **GREEN** ("BENIGN TRAFFIC").
    - The AI Analyst report explains that the low packet count and standard port indicate safe activity.

Scenario B: DDoS Attack Simulation

Simulate a malicious bot flooding the network.

1. **Input Parameters:**
   - Duration: 10 (ms) -> *Inhumanly fast*
   - Total Packets: 500 -> *High volume burst*
   - Length Mean: 0 -> *Empty payload*
   - Active Mean: 10
2. Click **INJECT PACKET**.
3. **Result:**
   - Status Banner turns **RED** ("! MALICIOUS ATTACK !").
   - **Visual Diagnostics Chart:** Indicates a "True Positive".
   - **AI Report:** The Groq AI will identify this behavior as a **DoS/DDoS** signature due to high frequency and short duration.

Scenario C: Random Stress Testing

1. Click the RANDOM SAMPLE button.
2. The system pulls a random row from the unseen "Test Set" (the 20% reserved during training).
3. This is useful to verify that the model works on data it hasn't generated itself.

Phase 3: Forensic Reporting

The system creates professional-grade audit documentation for security reviews.

1. After completing your analysis, click the White ⇩ **REPORT** button in the top navigation bar.
2. Choose a **Save** location.
3. Open the PDF. It is divided into three sections:
   - **Page 1 (Executive Summary):** Contains Project Metadata, Model Accuracy Metrics, and the Visual Confusion Matrix heatmap.
   - **Page 2 (Forensic Deep Dive):** Contains the exact raw header data of the last analyzed packet and the full text of the AI's forensic conclusion.
   - **Page 3 (System Audit Logs):** A complete transcript of every event from the "System Logs" terminal for compliance purposes.

---

6. Glossary

- **NIDS:** Network Intrusion Detection System.
- **Confusion Matrix:** A table layout that visualizes the performance of the algorithm (Safe predicted as Safe vs. Attacks predicted as Safe).

- **Flow Duration:** The length of time a connection was open (ms).
- **Forward Packets:** Data sent from the source to the destination.
- **Inference:** The process of the AI "thinking" about the data.

---

7. Troubleshooting

ISSUE	CAUSE	-SOLUTION-
System Status stuck on "WAITING"	Model not trained	Click the **INITIALIZE** button to train the brain before injecting packets.
API Error (401)	Missing Key	Ensure you pasted a valid Groq API key and clicked **SAVE**.
NameError: 'pd' is not defined	Missing Import	Update your `main.py` (Fixed in version 3.0). Ensure `import pandas as pd` is at the top.
CSV File Not Loading	Wrong format	Ensure the dataset downloaded from Kaggle is the **MachineLearningCSV** version, not the raw PCAP files.

---

8. Support & References

- **Repository:** ***https://github.com/hi-ashup/AI-Based-Network-Intrusion-Detection.git***
- **Data Source:** Canadian Institute for Cybersecurity (CIC-IDS2017)
- **License:** MIT Open Source License.
- **AI Engine:** Powered by Llama-3-70b-Versatile via **Groq Cloud**.
- **ML Engine:** Scikit-Learn Random Forest Classifier.

—**END**—