VoIP Traffic Capture and IP Geolocation Mapping

This repository contains a Python script for capturing VoIP traffic, retrieving geolocation data for IP addresses involved in the traffic, and an HTML/JavaScript file to visualize this data on a map.

Summary

Python Script (`capture_voip.py`):

1. **Capture VoIP Traffic**: Uses `pyshark` to capture live VoIP (SIP and RTP) packets from the specified network interface.
2. **Extract IP Addresses**: Analyzes captured packets to extract source and destination IP addresses.
3. **Get Geolocation Data**: Uses the IPinfo API to retrieve geolocation information (latitude and longitude) for the IP addresses.
4. **Save Data**: Stores the geolocation data in `geolocation_data.json` for use by the HTML file.

HTML/JavaScript File (`map_visualization.html`):

1. **Load Google Maps**: Uses Google Maps API to display a map.
2. **Fetch Geolocation Data**: Retrieves geolocation data from `geolocation_data.json`.
3. **Display Markers**: Places markers on the map at the locations specified by the geolocation data.

Prerequisites

1. **Python 3.x**: Ensure Python is installed on your system.
2. **Python Libraries**: `pyshark`, `ipinfo`
3. **Google Maps API Key**: Required for the HTML file to display maps.

Installation

Python Dependencies

1. Install the required Python libraries:

   ```
   pip install pyshark ipinfo
   ```

2. Obtain an IPinfo API token by signing up at IPinfo.io and replace `'your_ipinfo_access_token_here'` in `capture_voip.py` with your token.

Google Maps API Key

1. Obtain a Google Maps API key from Google Cloud Console.
2. Replace `YOUR_GOOGLE_MAPS_API_KEY` in `map_visualization.html` with your actual API key.

Usage

Running the Python Script

1. Save the Python code as `capture_voip.py`.
2. Run the script to start capturing VoIP packets and saving geolocation data:

   ```
   python capture_voip.py
   ```

   - The script will continuously capture VoIP traffic on the specified network interface (replace `'en0'` with your network interface if different) and save geolocation data to `geolocation_data.json`.

Serving the HTML and JSON Files

1. Ensure `geolocation_data.json` and `map_visualization.html` are in the same directory.
2. Use a simple HTTP server to serve the files:

   ```
   python -m http.server 8000
   ```

3. Open a web browser and navigate to `http://localhost:8000/map_visualization.html` to view the map with the geolocation data.

Notes

- **Network Interface**: The default network interface in the script is set to `'en0'` for macOS. Adjust this to match your network interface if necessary.
- **Legal and Ethical Use**: Ensure you have the necessary permissions and are compliant with all legal and ethical guidelines when capturing and analyzing network traffic.