Threat Intelligence Extraction Tool

This Python tool is designed to extract valuable threat intelligence from PDF-based threat reports. It uses natural language processing (NLP), regular expressions, and APIs to identify indicators of compromise (IoCs), tactics, techniques, and procedures (TTPs), malware names, threat actors, and targeted entities.

Features

1. **Extract Indicators of Compromise (IoCs):**
   - IP Addresses
   - Domains
   - Email Addresses
   - File Hashes (MD5, SHA1, SHA256)
2. **Identify Tactics, Techniques, and Procedures (TTPs):**
   - Extract TTPs using an AI model (Llama API).
3. **Detect Threat Actors:**
   - Identify threat actor groups or individuals mentioned in the report.
4. **Malware Analysis:**
   - Extract malware names from the report.
   - Enrich malware data with VirusTotal API to fetch hash values and associated tags.
5. **Identify Targeted Entities:**
   - Extract organizations (ORG) and geographical entities (GPE) using SpaCy.

Requirements

- Python 3.8 or higher
- Required Python packages:
  - re
  - requests
  - json
  - spacy
  - pdfminer.six

Setup

1. **Install Python Packages:**
   - python -m spacy download en_core_web_sm
   - pip install -r requirements.txt
2. **API Keys:**
   - To run this tool, you need LLAMA and Virus Total API Keys.
   - Replace LLAMA_API_KEY with your Llama API key.
   - Replace x-apikey in the VirusTotal API headers with your VirusTotal API key.
3. **Run the Program:**
   - python threat_intelligence_extractor.py

Input

- A PDF or text file containing a threat report.

Output

- Extracted threat intelligence data in JSON format, including:
    - IoCs
    - TTPs
    - Threat Actor(s)
    - Malware
    - Targeted Entities

Code Breakdown

1. Llama API Integration

- The llamaApiQuery function queries the Llama API to extract TTPs, malware names, and threat actors.
- The API requires a valid token passed via the Authorization header.

2. VirusTotal Reference

- The extractMalware function queries the VirusTotal API for hash details and tags related to the malware.

3. IoC Extraction

- Uses regular expressions to extract:
    - IP addresses
    - Domains
    - Emails
    - File hashes

4. NLP for Entity Extraction and Tokenization

- Utilizes SpaCy’s pre-trained en_core_web_sm model to identify organizations (ORG) and geographical entities (GPE).

5. PDF Text Extraction

- The pdfminer.six library extracts text content from the input PDF.

Usage

Provide the PDF file name when prompted:

Enter File Name (PDF or text file): threat_report.pdf

The extracted data will be displayed in JSON format:

```
{
  "Iocs": {
    "IPAddresses": ["192.168.1.1"],
    "Domains": ["example.com"],
    "Emails": ["test@example.com"],
    "FileHashes": ["d41d8cd98f00b204e9800998ecf8427e"]
  },
  "Ttps": {...},
  "ThreatActor(s)": [...],
  "Malware": [...],
  "TargetedEntities": [...]
}
```

Notes for Developers

1. Error Handling:
   - The program handles common API issues, such as timeouts or invalid responses, and prints error messages.
2. Extensibility:
   - You can add additional IoC patterns in the extractIocs function.
   - Extend the Llama API queries for custom data extraction.
3. Logging:
   - Informative messages ([INFO], [WARNING], [ERROR]) are printed to assist in debugging.
4. Performance Considerations:
   - Limit the input size for Llama API queries to 4,000 characters.