

Revenue Recon

B2B Threat & Opportunity Intelligence Suite

Automated Corporate Reconnaissance & Strategic Analysis

The **B2B Intel Suite** is a comprehensive intelligence gathering tool designed for sales engineers, security analysts, and business developers. It automates the process of digital due diligence by combining **Open-Source Intelligence (OSINT)** scanning, **Network Security Analysis**, and **Generative AI** to produce actionable strategic reports.

Table of Contents

- [Features](#)
 - [Architecture](#)
 - [Installation](#)
 - [Configuration](#)
 - [Usage](#)
 - [CLI Mode](#)
 - [Web Dashboard](#)
 - [Modules](#)
 - [API Reference](#)
 - [Contributing](#)
 - [License](#)
-

Features

Intelligence Gathering

- **Digital Footprint Mapping:** Automatically discovers official business websites, social media profiles, and digital assets using advanced OSINT techniques.
- **Tech Stack Detection:** Identifies the underlying technologies (CMS, Frameworks, Analytics) used by target companies.
- **Contact Extraction:** Scrapes public-facing emails and social media handles for lead generation.

Security Assessment

- **Port Scanning:** Performs non-intrusive scans on critical ports (21, 22, 80, 443, 3389) to identify potential vulnerabilities.
- **SSL/TLS Verification:** Validates SSL certificate chains, expiration dates, and issuer integrity.
- **DNS Analysis:** Checks for email security protocols (SPF, DMARC) to assess phishing risk.

AI-Powered Analysis

- **Executive Summaries:** Uses **Google Gemini AI** to synthesize raw technical data into executive-level narratives.
- **Strategic Proposals:** Automatically generates tailored marketing and security improvement proposals based on detected gaps.
- **Sentiment Analysis:** Analyzes public customer reviews to gauge brand reputation and operational weaknesses.

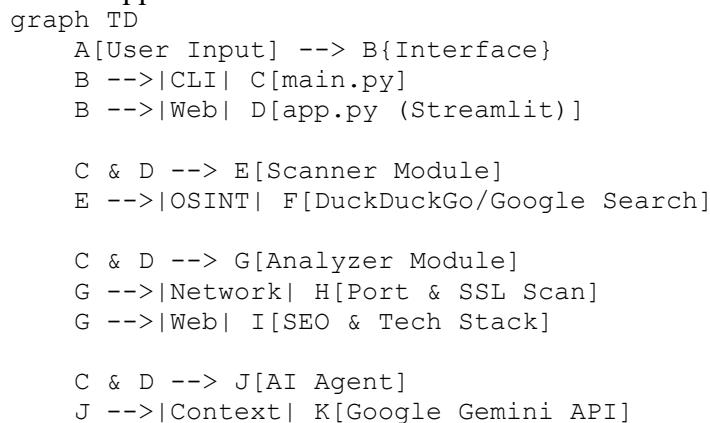
Reporting

- **PDF Generation:** Produces professional, branded PDF reports containing both high-level summaries and raw technical appendices.
 - **Risk Scoring:** Calculates a proprietary "Risk vs. Opportunity" score (0-100) for quick assessment.
-

Architecture

The project follows a modular architecture to separate data collection, analysis, and presentation layers.

Code snippet



C & D --> L[Reporter Module]
L --> M[PDF Report]

Installation

Prerequisites

- Python 3.10 or higher
- Git
- A Google Cloud API Key (for Gemini)

Step-by-Step Guide

1. Clone the Repository

Bash

```
git clone https://github.com/nanashi151/b2b-intel-suite.git  
cd b2b-intel-suite
```

2. Create a Virtual Environment

Bash

```
python -m venv venv  
# Windows  
venv\Scripts\activate  
# macOS/Linux  
source venv/bin/activate
```

3. Install Dependencies

Bash

```
pip install -r requirements.txt
```

⚙ Configuration

The application requires environment variables to function correctly, particularly for the AI integration.

1. Create a `.env` file in the root directory.
2. Add your API keys and configuration settings:

Ini, TOML

```
# .env file
GEMINI_API_KEY="your_google_gemini_api_key_here"
LOG_LEVEL="INFO"
USER_AGENT="B2B-Intel-Scanner/1.0"
```

Note: Never commit your `.env` file to version control. It is added to `.gitignore` by default.

█ Usage

Web Dashboard (Recommended)

Launch the interactive Streamlit dashboard for a visual experience.

Bash

```
streamlit run app.py
```

- **Access:** Open your browser to `http://localhost:8501`.
- **Functionality:** Enter a business name or URL to trigger a real-time scan and download PDF reports directly.

CLI Mode (Automation)

Run the tool directly from the terminal for quick scans or batch processing.

Bash

```
python main.py --target "Target Company" --location "City, Country"
```

Options:

- `--target`: Name of the business to scan.
 - `--location`: Geographic location to narrow down OSINT results.
 - `--output`: (Optional) Path to save the PDF report.
-

❖ Modules

`scanner.py`

Handles the initial reconnaissance and discovery.

- `find_business_url(name, location)`: Locates the official website using search heuristics.

`network_scanner.py`

Performs active and passive network analysis.

- `scan_common_ports(ip)`: Checks for open high-risk ports.
- `check_ssl_chain(url)`: Validates the certificate trust chain.

`analyzer.py`

Extracts business logic and marketing data.

- `detect_tech_stack(html)`: Identifies frameworks (e.g., React, WordPress).
- `extract_emails(text)`: Scraps contact information using regex patterns.

`ai_agent.py`

Interfaces with the LLM for high-level reasoning.

- `generate_executive_summary(data)`: Synthesizes scan results into a narrative.
- `analyze_sentiment(reviews)`: Processes customer feedback.

`reporter.py`

Compiles all findings into a structured document.

- `generate_pdf(data)`: Renders the final audit report using fpdf.
-



Contributing

Contributions are welcome! Please follow these steps to contribute:

1. **Fork the Project**
2. **Create your Feature Branch** (`git checkout -b feature/AmazingFeature`)
3. **Commit your Changes** (`git commit -m 'Add some AmazingFeature'`)
4. **Push to the Branch** (`git push origin feature/AmazingFeature`)
5. **Open a Pull Request**

Code Standards

- Follow **PEP 8** style guidelines.
 - Ensure all new functions have docstrings.
 - Run tests before submitting (if applicable).
-

License

Distributed under the MIT License. See `LICENSE` for more information.

Contact

Project Link: <https://github.com/carmelaidan/revenue-recon>

Email: carmelmendez1511@gmail.com

LinkedIn: <https://www.linkedin.com/in/carmel-aidan-mendez/>

Website: <http://carmelmendez.vercel.app/>

Phone Number: [\(+63\)9493292770](tel:+639493292770)