

# PortAtlas – Full Documentation Manual

**PortAtlas** is an advanced network scanning tool inspired by Nmap, designed for developers, sysadmins, and security analysts. It supports TCP/UDP scanning, service detection, OS fingerprinting (lightweight), and API integration. This manual covers installation, usage, and features.

## 1. Installation

- Clone the repository: `git clone https://github.com/abiola-samwel/PortAtlas-Scanner`
- Navigate to project: `cd portatlas`
- Install dependencies: `pip install -r requirements.txt`

## 2. Command-Line Usage

**Basic Syntax:**

`python test_scan.py <target> <ports> [options]`

**Options:**

Flag	Description
<code>--type syn udp tcp_connect</code>	Select scan type (SYN = stealth, UDP = datagram, TCP connect = full handshake).
<code>--banner</code>	Enable banner grabbing (fetch service details).
<code>--all</code>	Scan all 65,535 ports.
<code>--no-json</code>	Disable JSON output (print only table).
<code>--api-key KEY</code>	Provide API key (used in secure mode).
<code>--debug</code>	Enable debug logging.
<code>--ignore-errors</code>	Skip errors and continue scanning.

**Examples**

- Scan 1000 common TCP ports:

`python test_scan.py 192.168.1.1 1-1000`

- UDP scan for DNS:

`python test_scan.py 8.8.8.8 53 --type udp --banner`

- Full scan with banners and JSON output:

`python test_scan.py 8.8.8.8 1-65535 --banner`

### 3. API Usage

The backend provides a FastAPI server exposing scan endpoints.

**Run API:** `uvicorn backend.app.main:app --reload --host 127.0.0.1 --port 8000`

**Example Request:**

```
curl -X POST "http://127.0.0.1:8000/scan" -H "Content-Type: application/json" -d '{"target": "8.8.8.8", "ports": "53", "scan_type": "udp"}'
```

### 4. Key Features

1. Comprehensive port scanning (TCP/UDP, 1–65535).
2. Banner grabbing for service fingerprinting.
3. Lightweight OS detection (based on banner evidence).
4. Stealth scanning (SYN, FIN, NULL modes).
5. Error handling with `--ignore-errors` option.
6. Security layers: API key (for online), audit logs, rate limiting.

### 5. Output Formats

- CLI: Colored table + JSON summary (optional).
- API: JSON response including target, ports, services, banners, and OS guess.
- Logs: Saved to `logs/` folder (rotates monthly).