

Minimal secure public "ping" API

This project runs a small Express server that responds with **ping** for any route and method. It includes safe defaults (Helmet, CORS, rate limiting, small body limits, logging, compression) and can optionally expose a public URL using a tunnel so others on the internet can query it while it runs on your Mac.

Quick start

1. Install dependencies:

```
npm install
```

2. Start locally:

```
npm run dev  
# or  
npm start
```

Visit <http://localhost:8080/ping>.

3. Expose publicly (optional):

```
npm run expose
```

The console will print a **Public URL**:. Share that URL. While the server runs, anyone can query it.

Security notes

- Defaults are conservative: only GET/HEAD are allowed cross-origin by default; body size limited to 10kb; rate limit of 60 requests/minute per IP.
- To allow broader access (e.g., POST), set **CORS_ORIGIN=*** in your environment.
- This is still an internet-exposed endpoint. Avoid adding stateful or file-system operations unless you fully validate and authenticate requests.
- Prefer running behind a random tunnel URL. Disable **ENABLE_TUNNEL** when not in use.

Configuration

Environment variables (can be placed in a **.env** file). See **env.sample** for a template.

- **PORT** (default **8080**) – port to listen on
- **HOST** (default **0.0.0.0**) – listen interface
- **CORS_ORIGIN** (default **GET_ONLY**) – **GET_ONLY**, *****, or comma-separated allowed origins
- **RATE_LIMIT_WINDOW_MS** (default **60000**)

- `RATE_LIMIT_MAX` (default `60`)
- `BODY_LIMIT` (default `10kb`)
- `AUTH_TOKEN` – if set, require `Authorization: Bearer <token>` for all requests
- `TRUST_PROXY` (default `1`) – trust proxy hops for correct client IPs behind tunnels
- `ENABLE_TUNNEL` (default `0`) – set to `1` to enable localtunnel
- `TUNNEL_SUBDOMAIN` – optional preferred subdomain
- `GEOCODER_BASE_URL` – base URL for a Nominatim-compatible geocoder used by `/geocode/bbox` (default `https://nominatim.openstreetmap.org`)

Example responses

```
{
  "message": "ping",
  "method": "GET",
  "path": "/anything",
  "query": { "foo": "bar" },
  "headers": { "user-agent": "..." },
  "timestamp": "2025-01-01T00:00:00.000Z"
}
```

Scripts

- `npm start` – start server
- `npm run dev` – start with auto-reload
- `npm run expose` – start and open a public tunnel

Geocoding Endpoint

- `GET /geocode/bbox?q=<address>`: Returns the latitude/longitude for the best match and the bounding box for the address.

Example:

```
# URL-encoded address (works in any shell)
curl "http://localhost:8080/geocode/bbox?q=1600%20Pennsylvania%20Ave%20NW%2C%20Washington%2C%20DC"

# or let curl encode it for you
curl -G --data-urlencode "q=1600 Pennsylvania Ave NW, Washington, DC" http://localhost:8080/geocode/bbox
```

Response:

```
{
  "query": "1600 Pennsylvania Ave NW, Washington, DC",
  "provider": "nominatim",
}
```

```
"center": { "lat": 38.897675, "lon": -77.036547 },  
"bbox": { "south": 38.897, "west": -77.037, "north": 38.898, "east":  
-77.036 }  
}
```

Notes:

- The default provider is OpenStreetMap Nominatim. You can point to your own Nominatim instance by setting `GEOCODER_BASE_URL`.
- If `AUTH_TOKEN` is set, include `Authorization: Bearer <token>` in requests.
- Default CORS allows only GET/HEAD from anywhere. This endpoint is GET-only and works with the default.
- Errors: `400` for missing/too-long query, `404` if not found, `502` if provider is unavailable, `504` for timeouts.