Checkmate HttpServer API v1 – Documentation

# 1. Overview

Checkmate HttpServer is a lightweight HTTP server built on NanoHTTPD for Android, providing RESTful endpoints to control device features (camera, audio, streaming, recording, GPS) and to serve recorded media with range requests. It also exposes an endpoint to trigger in-app Magisk module updates.

# 2. Authentication

To secure access, the server supports API key authentication. Set an environment variable CHECKMATE\_API\_KEY when starting the server. All requests (except OPTIONS and HEAD) must include:

* • X-API-Key: your\_api\_key

# 3. CORS and HTTP Headers

The server automatically handles CORS preflight requests (OPTIONS) and responds with the following headers:

* • Access-Control-Allow-Origin: \*
* • Access-Control-Allow-Methods: GET, POST, PUT, DELETE, OPTIONS, HEAD
* • Access-Control-Allow-Headers: Content-Type, Range, X-API-Key

# 4. Error Handling

* • 400 Bad Request: Invalid parameters
* • 401 Unauthorized: Missing or invalid API key
* • 404 Not Found: Endpoint or resource not found
* • 416 Range Not Satisfiable: Invalid byte range
* • 500 Internal Server Error: Unexpected errors

# 5. Request & Response Format

All endpoints consume and produce JSON. Include Content-Type: application/json for POST requests. Responses follow this structure:

{  
 "field": "value",  
 "field2": true,  
 ...  
}

# 6. Endpoints

| Method | Endpoint | Body | Description |
| --- | --- | --- | --- |
| GET | /api/v1/system/info | - | Returns device info: model, serial, battery, app version |
| GET | /api/v1/camera/status | - | Current camera (Front/Back) |
| POST | /api/v1/camera/set | {"type":"Front"} | Set camera type |
| GET | /api/v1/audio/status | - | Audio enabled status and type |
| POST | /api/v1/audio/set | {"enable":true} | Enable or disable audio |
| GET | /api/v1/streaming/status | - | Streaming status |
| POST | /api/v1/streaming/set | {"action":"start"} | Start or stop streaming |
| GET | /api/v1/recording/status | - | Recording status |
| POST | /api/v1/recording/set | {"action":"start"} | Start or stop recording |
| GET | /api/v1/playback/list | - | List MP4 recordings |
| POST | /api/v1/playback/download | {"filename":"clip.mp4"} | Download recording with range support |
| GET | /api/v1/gps/status | - | GPS enabled status and coordinates |
| POST | /api/v1/gps/set | {"action":"start","interval":60} | Start/stop GPS or set interval |
| POST | /api/v1/app/update | {"url":"<module-zip-url>"} | Trigger download & install of a new app module |

# 7. Example Usage

## Trigger a Magisk Module Update

Request:

POST /api/v1/magisk/update HTTP/1.1  
Host: localhost:8080  
Content-Type: application/json  
X-API-Key: your\_api\_key  
  
{  
 "url": "https://example.com/mymodule-3.380.zip"  
}

Response:

{  
 "status": "Magisk update triggered"  
}