

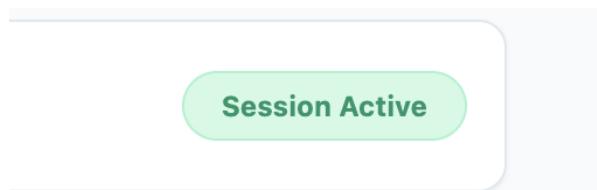
User Manual: Geotab Keyless Manager

1. Authorization

The screenshot shows the Geotab Keyless Manager interface. At the top left is the Telefonica Tech logo. Next to it is the title "Geotab Keyless Manager". On the far right, a red rounded rectangle contains the text "No Active Session". Below the title, there's a section labeled "1. Authorization" with three input fields: "telefonica_empleados", "daniel.garciafernandez@telefonica.com", and a password field starting with ".....". To the right of these fields are three buttons: "Deploy Selected" (green), "Revoke Selected" (red), and "Sync All" (blue). Further right is a "Import CSV" button. Below these buttons is a "Quick Add" bar with fields for "GO9 Serial" (containing "GO9 Serial") and "Description", followed by an "Add" button. At the bottom of the authorization section is a blue "Connect & Save" button. To the right of the authorization section is a table header with columns: "GO9 SERIAL", "DESCRIPTION", "KEY ID / REF", and "ACTION".

To interact with the Geotab Keyless API, you must first authenticate:

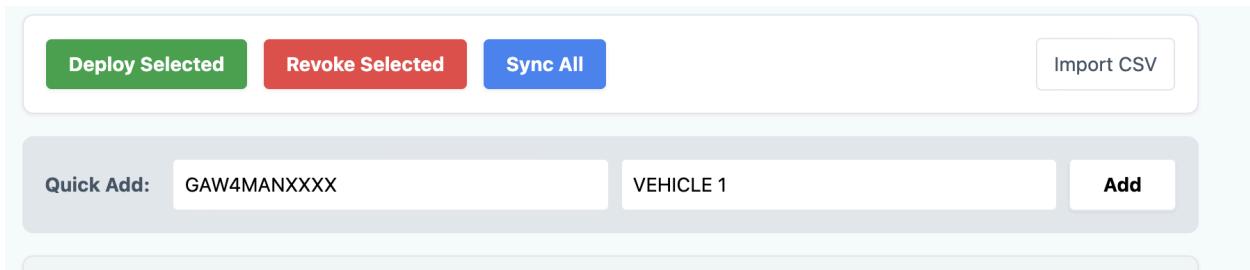
- **Database:** Your Geotab database name (Tenant).
- **User/Pass:** Your Geotab credentials.
- **Persistent Settings:** The tool saves your Database and Email locally so you don't have to re-type them every time you open the app.
- **Indicator:** A green "Session Active" badge will appear upon successful login.



2. Fleet Management

Adding Vehicles

- **Quick Add:** Use the gray bar to add a single GO9 serial number and a friendly description.
- **Import CSV:** Click "Import CSV" to upload a list. The format should be `SerialNumber, Description` (no headers).
- **Remove Locally:** Click the red × to remove a vehicle from your local view (this does *not* delete anything in the Geotab cloud).



Syncing Keys

The **Sync All** or individual **Sync** buttons will query the Geotab API to see if a "Stored Virtual Key" already exists on the device. If found, it will retrieve and store the `virtualKeyId` and `userReference` in your local list.

<input type="checkbox"/>	G09 SERIAL	DESCRIPTION	KEY ID / REF	ACTION
<input type="checkbox"/>	GAW4MAN8Y XXXXXXXXXX	0000000000	00000000-0000-0000-0000-000000000000 Geotab Device Identifier - (00000000)	Sync X
<input type="checkbox"/>	GAH1D6VAY XXXXXXXXXX	0000000000	00000000-0000-0000-0000-000000000000 Geotab Device Identifier - (00000000)	Sync X
<input type="checkbox"/>	GAP9JVB63 XXXXXXXXXX	0000000000	00000000-0000-0000-0000-000000000000 Geotab Device Identifier - (00000000)	Sync X
<input type="checkbox"/>	GAVDJKM8E XXXXXXXXXX	0000000000	00000000-0000-0000-0000-000000000000 Geotab Device Identifier - (00000000)	Sync X
<input type="checkbox"/>	GAX206SPF XXXXXXXXXX	0000000000	00000000-0000-0000-0000-000000000000 Geotab Device Identifier - (00000000)	Sync X

3. Virtual Key (VK) Deployment

Customizing the Body

The **VK Body Template** (JSON) defines what the keys will do. You can modify:

- `tapCardSerialNumbers`: The list of physical NFC tags allowed to start the car.
- `permissions`: Lock, Unlock, IgnitionEnable, etc.
- `userReference`: The name that will appear in Geotab reports.
- `Timestamps`: The tool automatically calculates a 1-year validity from the moment you load the page.

2. VK Body Template

```
{  
    "isStoredVirtualKey": true,  
    "tapCardSerialNumbers": [  
        "7CF22JDPA1",  
        "7CF22JCTA1"  
    ],  
    "userReference": "Master Key",  
    "beginningTimestamp": 1766532097237,  
    "endingTimestamp": 1798068097237,  
    "permissions": [  
        "Lock",  
        "Unlock",  
        "IgnitionInhibit",  
        "IgnitionEnable"  
    ],  
    "privileges": [  
        "CheckinOverride"  
    ],  
    "endBookConditions": [  
        "CheckinOverride"  
    ]  
}
```

Bulk Actions

1. **Select:** Use the checkboxes to select specific vehicles or "Select All."
2. **Deploy:** Hits the API to create the keys defined in your JSON template.
3. **Revoke:** If a **Key ID** is present in the table, this will call the DELETE API to remove that specific virtual key from the hardware.

4. Auditing & Logs

Every time a **Create** or **Delete** action is performed, a record is added to the **Audit Change Logs** at the bottom of the screen.

- **Log Data:** Includes the Timestamp, the User (email), the Serial Number, and the exact JSON parameters sent to the API.
- **Download:** Use the **Download Log File** button to export the entire history as a **.txt** file for your internal compliance reports.

Audit Change Logs

[Download Log File \(.txt\)](#)

No logs available.