Powerfleet for Vehicles
Keyless Entry Overview

Exclusively created for

SIXT

Table of Contents

1. Introduction

This document provides an overview of the KVG (***Keyless Vehicle Gateway)*** and RentalFleet long loop(cellular) and short loop (BLE) feature. The feature allows an API service or mobile device to communicate with a KVG via cell network or BLE to transmit a command to trigger a recorded key fob command from the device. **The recorded key fob command is referred to as a *telegram* within the system.**

This is made possible via the use of KVG firmware (upgradeable over the air, managed by Powerfleet), a mobile application (both iOS and Android, managed by the customer), a KVG mobile app SDK (managed and distributed by Powerfleet), and backend API services referred to as RentalFleet.

The following sections will detail the overview of the keyless entry feature from beginning to end which includes:

- Telegram capture process via a second key fob
- KVG downloading and storing the telegrams
- Sending the lock/unlock command to the KVG via long or short loop
- KVG triggering and transmitting the telegram to the vehicle via ultrahigh frequency

2. Key Fob Recording Location

Second key fobs of vehicles are typically and suggested to be stored in a central repository. This is best practice for inventory and tracking purposes for the customer. Powerfleet can be collocated within the building of the central repository as a convenient location to be sent, retrieve, and request the second key fob. This allows Powerfleet to quickly record the key fob data required to issue lock and unlock commands over long or short loop methods. **Figure 1** below outlines the process.

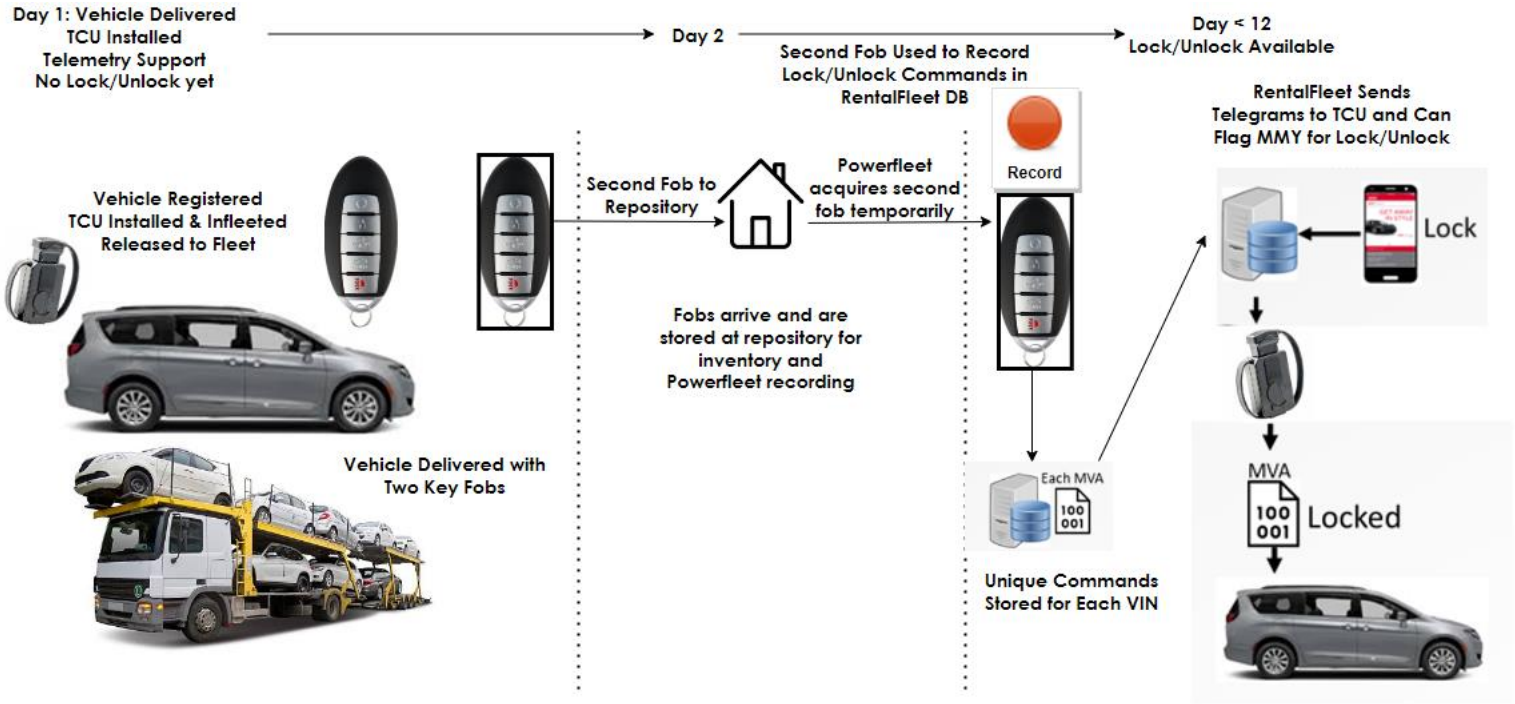

Figure 1

3. Telegram Recording & Upload Process

Automation is used to alternately depress key fob lock and unlock buttons while receivers are used to record and tag lock and unlock radio frequency transmissions.

Each key fob button press transmission is recorded and written to a backend database using proprietary software installed at each station. Each station is an RF shielded box to prevent interference or lost commands from the key fob. Multiple station setups are used to allow for multiple concurrent recordings and the station set ups can record any form factor of any key fob. Data is validated along the way to ensure no partial transmissions or duplicates are recorded.

A two-step process is used to transfer the recorded telegrams to the RentalFleet production environment. Step 1 uploads recorded telegrams from the recording stations at the repository to the RentalFleet database. Step 2 moves the telegrams to the production server and creates encrypted telegram set files.

At in-fleet or upon a periodic diagnostic record, RentalFleet pushes up to 4 encrypted telegram set files to the KVG.

The KVG will maintain at most 4 telegram set files (256 telegrams, 64 telegrams per file) at any given time and will automatically download more files as needed.

Initial Upload

As mentioned above, the telegrams are recorded and validated locally. A typical recording will use 256 key presses and all recordings from the presses will be validated and confirmed. Invalid recordings from each key press will be automatically discarded if necessary. The validated key presses are then transferred to the Powerfleet hosted environment over a secure VPN tunnel through a RESTful API service call.

Production Transfer

Once the telegrams have been databased, a batch process is used to synchronize all collected telegrams to the production server. This involves the Powerfleet hosted server making a RESTful API call to the production hosted server. This server process then creates a set of encrypted (AES-256) files that contain the telegrams. Each file contains at most 64 telegrams (32 lock, 32 unlock, alternating between lock and unlock), for a max total of 16 files for sets of up to 1024 telegrams. This process of data being transferred is summarized below in **Figure 2**.

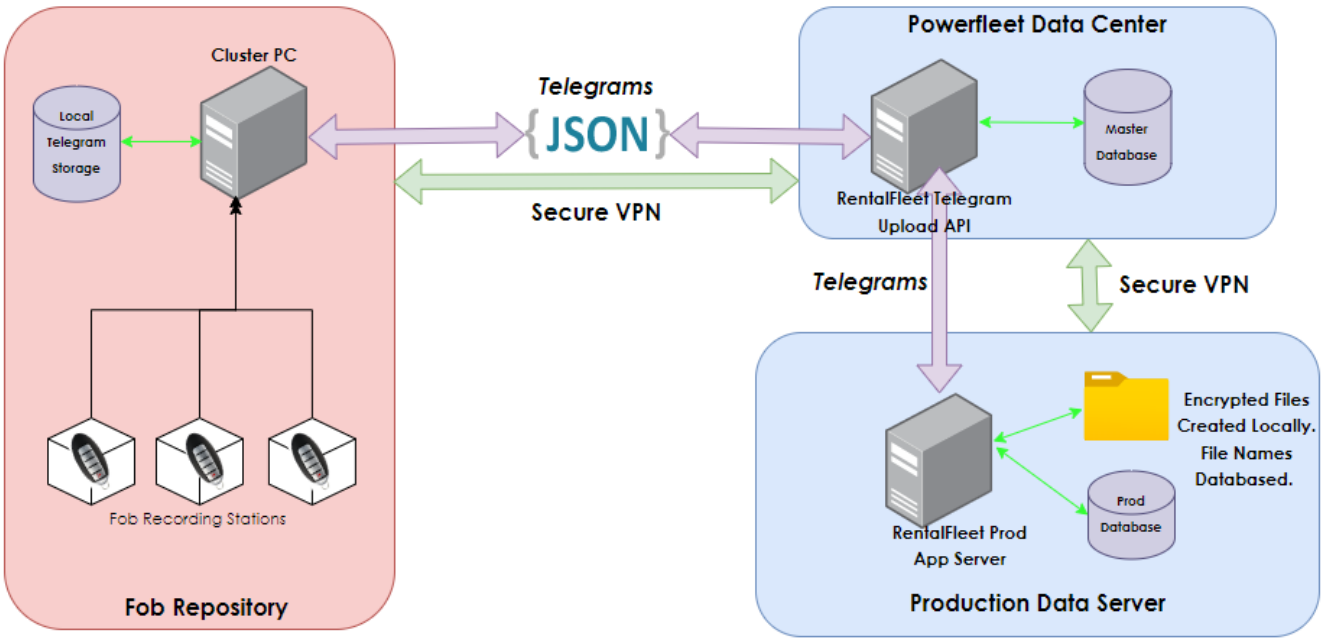

Figure 2

4. RentalFleet Telegram File Management

The RentalFleet application represents the middleware that bridges the KVG to the production server and the end customer operations platform. RentalFleet is comprised of a collection of services that allow the KVG to upload data and can send downlink messages to the KVG directly. RentalFleet also contains all the business logic needed for managing the FOTA process, telegram file download process, and the general business logic required to forward telemetry data to the end customer operations platform. The telegram file download process is a prerequisite to being able to perform successful lock/unlock over long or short loop. For the short loop process, once the telegram files are downloaded on the KVG, an API call is needed to generate one-time use tokens that act as a nonce for issuing commands. This is covered in a section below.

Figure 3 below describes RentalFleet and KVG telegram file management. The KVG will upload certain information that helps RentalFleet determine how many files are stored on the KVG and how many telegrams remain unused. RentalFleet then makes the decision to tell the KVG to download more files securely from the file server, using TLS 1.2 to perform the download. RentalFleet keeps track of how many telegrams exist on the device and always tries to keep between 192 and 256 unused telegrams on board (4 files).

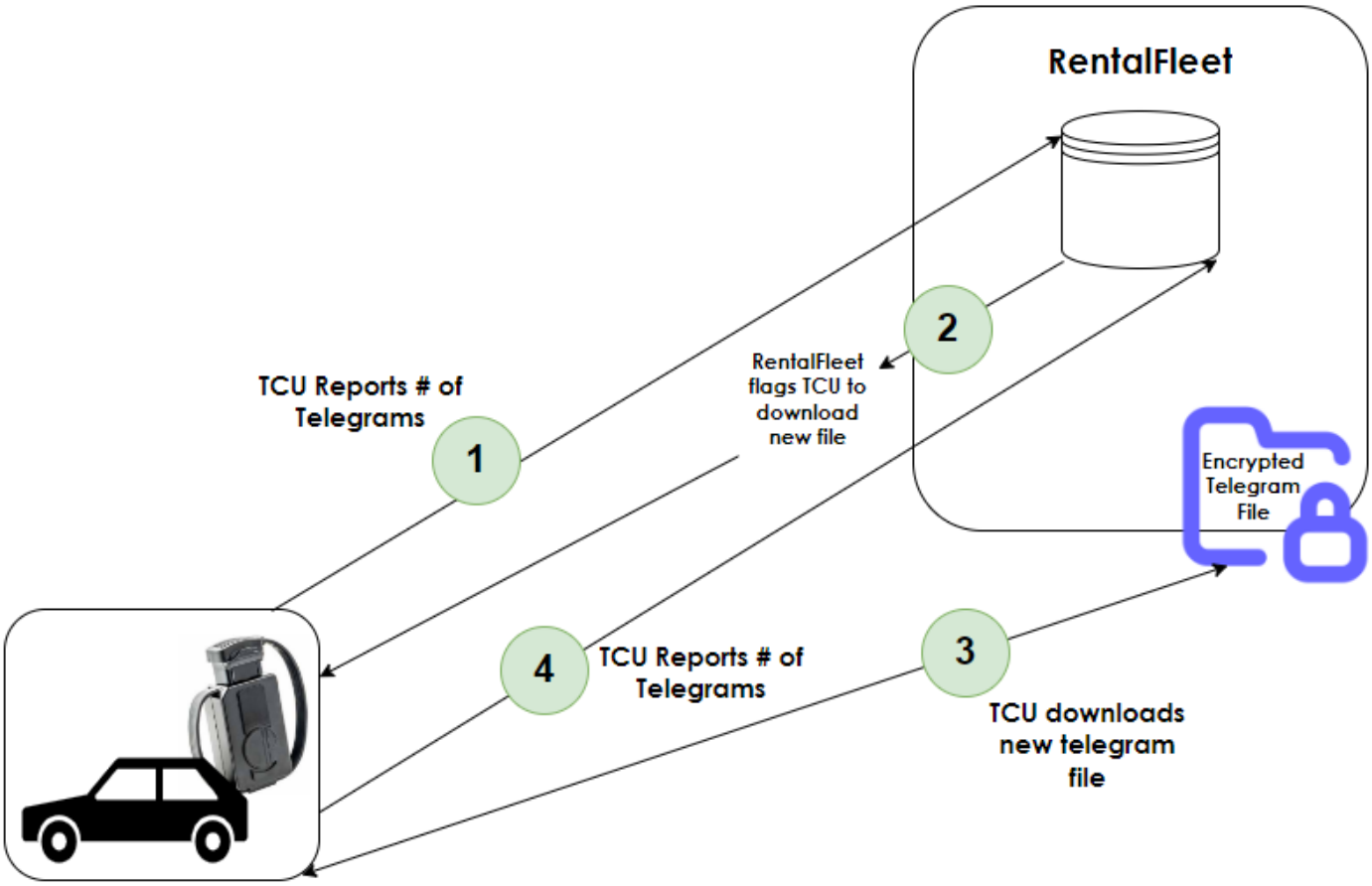

Figure 3

5. Long Loop (Cellular) Process Overview

The long loop process refers to using the cell network to communicate with the KVG to trigger a telegram on the device to transmit lock or unlock to the vehicle.
This can be accomplished in a few ways.

- Direct API service designed by the customer
- Mobile application to send API call
- RentalFleet website to directly send command

Once the telegrams are downloaded on the KVG, they can be triggered by any of the processes (using short loop via BLE is the only method that applies an additional layer of security via the SDK).

As mentioned in the sections above, the telegrams used to lock/unlock the vehicle are stored on the device. Sending a command to the KVG simply tells the device to trigger the next lock or unlock telegram in order. Every command sent of lock and unlock that is received by the KVG will return a command confirmation receipt with a response value to identify whether the lock or unlock was successful, failed, command error, no telegrams available, and other possible responses.

Once the command is triggered via long loop, < 3 seconds later the vehicle will lock or unlock.

6. Short Loop (BLE) Process Overview

The short loop process utilizes a mobile device with BLE enabled that can connect to a KVG using an SDK provided by Powerfleet. The SDK is meant to both obfuscate and provide seamless integration for third party mobile apps to be able to pair and communicate to the KVG devices. The SDK does not store, calculate, encrypt/decrypt or otherwise render data. All it does is pass what is given by the mobile app to KVG and vice versa. The app that contains the SDK never talks directly to the KVG. This solution covers both iOS and Android devices.

The KVG BLE SDK is used to perform three key features:

- Pair with target KVG via BLE
- Send command to target KVG
- Relay KVG response to calling app

To send a command and trigger lock and unlock via BLE securely, a **token** is used to authenticate a mobile user so that they can securely request commands over BLE to the KVG. After pairing the mobile device to the KVG, the KVG will validate the token and if it passes validation will issue the requested command to the vehicle's BCU using the device telegram files in has in memory. The token data is comprised of information specific to a vehicle (VIN), the mobile phone (UUID), KVG (MAC address), a window of time (start and end) that the token should be valid for, a rolling code that keeps the token both unique per user and sequential, along with information that is secret and

considered random. All these pieces of information are concatenated together and encrypted with the same encryption key that the KVG uses to upload data or RentalFleet uses to send downlink commands. Tokens can be cached on the mobile device and used at a later date, provided the date falls within the active period for when the token was requested.

Once the command is triggered via the mobile app using BLE, < 2 seconds later the vehicle will lock or unlock.

See **Figure 4** below illustrating the interconnections from client to device for Long Loop and Short Loop. This diagram is the overview of the keyless entry solution.

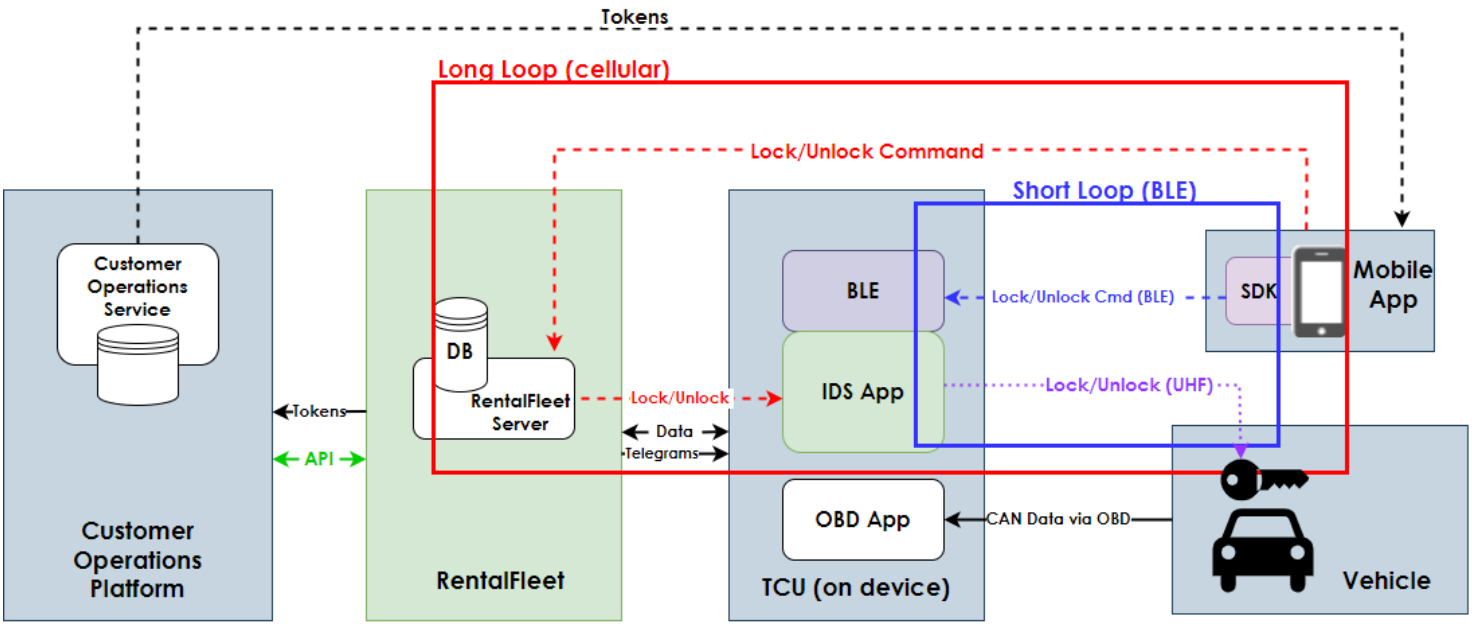

Figure 4