**Инструкция по тестированию EGTS-модуля**

Для тестирования EGTS-модуля передачи данных в РНИС пользователю необходимо выполнить последовательно 3 нижеописанных пункта:

# Скачать ПО

git clone https://github.com/SatisSoft/egts-debugger.git

# Установить скрипт

Для работы ПО необходим python3.6

Команда установки необходимых модулей:

sudo pip3.6 install crcmod

Команда запуска отладочного скрипта:

python3.6 start\_server\_debugger.py

По умолчанию скрипт слушает данные порта localhost:9090 и ожидает прием 10 EGTS-пакетов от клиента. Вместо данных параметров можно передать в скрипт собственные значения. К примеру, чтобы задать ожидаемое число пакетов равным 2, запуск скрипта осуществляется следующей командой:

python3.6 start\_server\_debugger.py -n 2

При необходимости авторизации в РНИС с применением подзаписи EGTS\_SR\_DISPATCHER\_IDENTITY, необходимый DISPATCHER\_ID задается при помощи флага «-d». К примеру, если DISPATCHER\_ID = 1234:

python3.6 start\_server\_debugger.py -d 1234

Для получения справки используется следующая команда:

python3.6 start\_server\_debugger.py -h

# Протестировать передачу данных

Пользователю необходимо осуществить следующую последовательность действий:

* Запустить скрипт (см. пункт 2).
* Запустить передачу данных по EGTS из собственного источника. В терминале, где запущен скрипт, будут отображаться логи.
* В случае если скрипт получит ожидаемое количество EGTS-пакетов (см. пункт 2), выполнение скрипта успешно завершается. Иначе, выполнение скрипта завершается с ошибкой.
* При успешном завершении работы скрипта необходимо свериться с логами и убедиться, что значения распарсенных параметров соответствуют ожидаемым.

Варианты завершения теста:

1) Ошибка парсинга EGTS.

Текст ошибки:

ERROR. EGTS connection test failed: error parsing EGTS packet. Error code {код ошибки}. {описание ошибки}

Коды и описание ошибок соответствуют информации, указанной в таблице №14 «Коды результатов обработки» протокола EGTS (Приказ Минтранса №285 <https://rnis.mos.ru/login/pdf/45.pdf>)

Выполнение скрипта прерывается.

2) Некорректный тип первого пакета.

Текст ошибки:

ERROR. EGTS connection test failed: The first packet must be EGTS\_SR\_DISPATCHER\_IDENTITY

Выполнение скрипта прерывается.

3) Неизвестная ошибка.

Текст ошибки:

ERROR. EGTS connection test failed: {описание ошибки}

соо

Выполнение скрипта прерывается.

4) Пришел только пакет авторизации, после чего соединение было разорвано.

Текст ошибки:

ERROR. EGTS connection test failed: received only auth packet

Выполнение скрипта прерывается.

5) Никакие данные не были получены.

Текст ошибки:

Error: received no data

Выполнение скрипта прерывается.

6) Некорректный Dispatcher Type:

Текст ошибки:

Dispatcher Type must be equal to 0. Currently it is equal to {описание ошибки}

ERROR. First packet is incorrect.

Выполнение скрипта прерывается.

7) Некорректный Dispatcher ID:

Текст ошибки:

Expected Dispatcher ID = 1234 but got {описание ошибки}

ERROR. First packet is incorrect.

Выполнение скрипта прерывается.

8) Получено меньше ожидаемых N пакетов.

Текст ошибки:

ERROR. Received only {число полученных пакетов} packets, expected {число ожидаемых пакетов} packets

Выполнение скрипта прерывается.

8) Успешное соединение.

Тест соединения считается успешным, если были получены пакет авторизации и как минимум 1 пакет с навигационными данными. После получения ожидаемого числа пакетов выполнение скрипта прерывается.

Текст сообщения:

SUCCESS. EGTS connection test succeeded. Received {число полученных пакетов} packets.

Please check in logs if data in packets is correct

Текст сообщения об успешно распарсенном пакете:

First packet is correct: Packet ID: 1; Packet Type: 1; records: [{RecNum: 1, sst: 1, subrecords: [{Type: 5, dt: 0, did: 1234}]}]

Received egts packet: Packet ID: 0; Packet Type: 1; records: [{RecNum: 0, sst: 2, ID: 239, subrecords: [{Type: 16, vld: True, ntm: 1533570258000, lat: 55.62752532903746, long: 37.782409656276556, speed: 0, dir: 178, busy: 0, src: 0}]}]

, где

* Packet ID – идентификатор пакета транспортного уровня;
* Packet Type – тип пакета транспортного уровня;
* Records – список записей;
* RecNum – номер записи;
* sst (source service type) – идентификатор типа Сервиса-отправителя, сгенерировавшего данную запись.
* subrecords – список подзаписей;
* Type – тип подзаписи;
* dt (dispatcher type) – тип диспетчера;
* did (dispatcher id) – уникальный идентификатор диспетчера;
* ID – идентификатор БНСТ;
* vld – битовый флаг, признак «валидности» координатных данных (1 – данные «валидны», 0 – «невалидные» данные);
* ntm – время навигации (Unix time);
* lat – широта
* long – долгота
* speed – скорость в км/ч;
* dir – направление движения;
* busy – статус «свободен/занят» для такси/каршеринг (1-занят, 0-свободен);
* src – определяет источник (событие), инициировавший посылку данной навигационной информации.

Пример логов при успешном завершении тестирования:

python3.6 start\_server\_debugger.py -n 4

First packet is correct: Packet ID: 1; Packet Type: 1; records: [{RecNum: 1, sst: 1, subrecords: [{Type: 5, dt: 0, did: 1234}]}]

Received egts packet: Packet ID: 0; Packet Type: 1; records: [{RecNum: 0, sst: 2, ID: 239, subrecords: [{Type: 16, vld: True, ntm: 1533570258000, lat: 55.62752532903746, long: 37.782409656276556, speed: 0, dir: 178, busy: 0, src: 0}]}]

Received egts packet: Packet ID: 0; Packet Type: 1; records: [{RecNum: 1, sst: 2, ID: 239, subrecords: [{Type: 16, vld: True, ntm: 1533570258000, lat: 55.62752532903746, long: 37.782409656276556, speed: 0, dir: 178, busy: 0, src: 0}]}]

Received egts packet: Packet ID: 0; Packet Type: 1; records: [{RecNum: 2, sst: 2, ID: 239, subrecords: [{Type: 16, vld: True, ntm: 1533570258000, lat: 55.62752532903746, long: 37.782409656276556, speed: 0, dir: 178, busy: 0, src: 0}]}]

SUCCESS. EGTS connection test succeeded. Received 4 packets.

Please check in logs if data in packets is correct.