

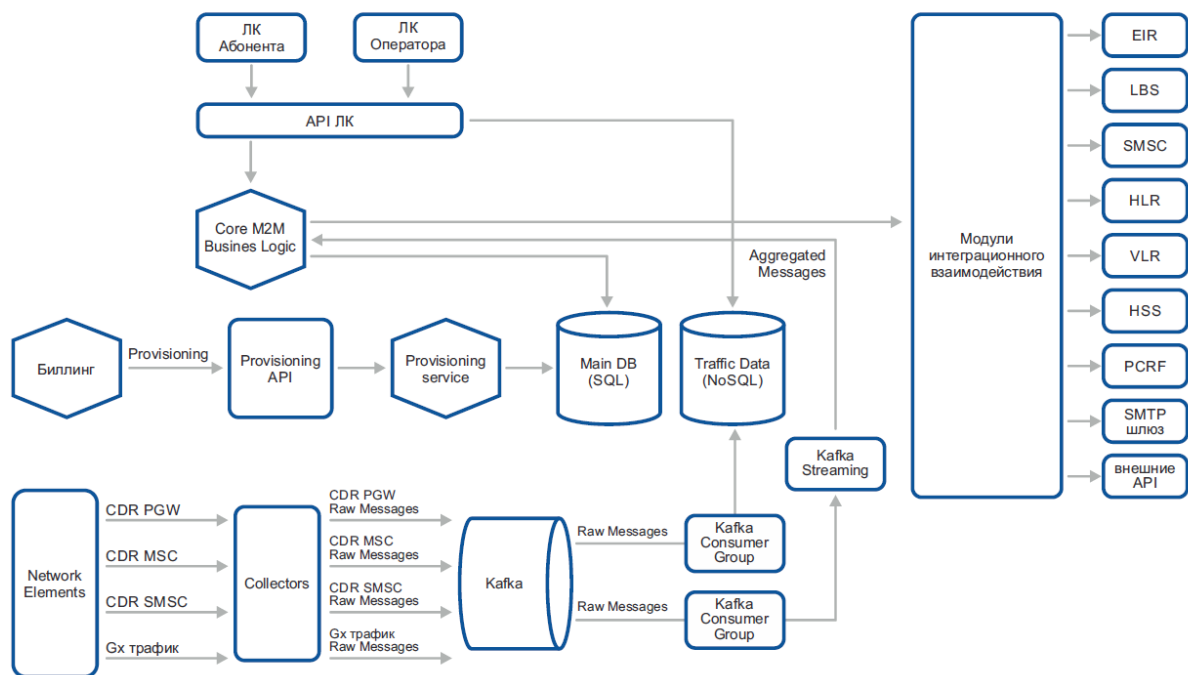


# **M2M платформа**

**Руководство администратора**

**ООО «ТЕНЕТ»**

## Упрощенная структурная схема M2M платформы



### Состав используемого программного обеспечения:

- Основная база данных: PostgreSQL
- База данных трафика: ClickHouse
- Шина данных: Kafka

### Управление модулями системы

#### 1. Kafka

Kafka работает в тандеме с Zookeeper

1.1. Скрипт управления общий для Kafka и для Zookeeper  
`systemctl start|stop|status kafka.service`

Проверить можно командами

Для Zookeeper: `netstat -tulpn | grep :2181`

Для Kafka: `netstat -tulpn | grep :9092`

1.2. Каталог инсталляции

`/opt/kafka/kafka_2.11-0.11.0.0`

`/opt/zookeeper/zookeeper-3.4.9`

1.3. Логи

`/var/log/kafka`

`/var/log/zookeeper`

Логи контроллера, сервера кафки и др.

`/opt/kafka/kafka_2.11-0.11.0.0/logs`

#### 2. Модули взаимодействия с Kafka

##### 2.1. Core

Модуль содержит классы доменных моделей передаваемых сообщений, утилиты со статическими методами преобразования данных и др. Это модуль является служебной библиотекой и используется другими модулями.

##### 2.2. PGWCDRAggregator

Модуль принимает поток сырых PGW\_CDR, а также поток списка устройств m2m (см. DeviceStorageConnector), фильтрует и пропускает только PGW\_CDR, принадлежащие

устройствам из списка. После фильтрации агрегирует значения и результат пишет в другую результирующую очередь, которая используется модулем проверки срабатывания триггеров(см `AggregatedMobileTrafficManager`).

2.2.1. Скрипт управления:

```
systemctl start|stop|status m2m_databus_pgw_cdr_aggregator.service
```

2.2.2. Каталог инсталляции:

```
/opt/m2m/databus-platform/PGWCDRAggregator/
```

2.2.3. Логи

```
/var/log/m2m/databus-platform/PGWCDRAggregator/
```

### **2.3. AggregatedMobileTrafficManager**

Принимает данные, агрегированные и отфильтрованные модулем `PGWCDRAggregator` `PGW_CDR`, записывает их блоками в таблицу БД устройств (Oracle), вызывает хранимую процедуру проверки соответствия predefined действиям (срабатывания триггеров), при наличии которых формирует список действий (отправка email или sms и отправляет действия в соответствующие очереди).

2.3.1. Скрипт управления:

```
systemctl start|stop|status m2m_databus_aggregated_traffic_manager.service
```

2.3.2. Каталог инсталляции:

```
/opt/m2m/databus-platform/AggregatedMobileTrafficManager/
```

2.3.3. Логи:

```
/var/log/m2m/databus-platform/AggregatedMobileTrafficManager/
```

### **2.4. AllowedDevices\_PGW\_CDR\_Filter**

Модуль принимает поток сырых данных в формате `PGW_CDR`, а также поток списка устройств `m2m` (см. `DeviceStorageConnector`), фильтрует их и пропускает только данные (в формате `PGW_CDR`), принадлежащие устройствам из списка и отправляет в результирующую очередь для модуля записи отфильтрованных (но не агрегированных) данных в ClickHouse (см. `FilteredRawPGWCDRPusherToDB`).

2.4.1. Скрипт управления:

```
systemctl start|stop|status m2m_databus_allowed_devices_pgw_cdr_filter.service
```

2.4.2. Каталог установки:

```
/opt/m2m/databus-platform/AllowedDevicesPGWCDRFilter/
```

2.4.3. Логи:

```
/var/log/m2m/databus-platform/AllowedDevicesPGWCDRFilter/
```

### **2.5. FilteredRawPGWCDRPusherToDB**

Принимает отфильтрованные модулем `AllowedDevices_PGW_CDR_Filter` данные в формате `PGW_CDR` и записывает их в БД ClickHouse.

2.5.1. Скрипт управления:

```
systemctl start|stop|status m2m_databus_filtered_raw_pgw_cdr_pusher_to_db.service
```

2.5.2. Каталог установки:

```
/opt/m2m/databus-platform/FilteredRawPGWCDRPusherToDB/
```

2.5.3. Логи:

```
/var/log/m2m/databus-platform/FilteredRawPGWCDRPusherToDB/
```

### **2.6. DeviceStorageConnector**

Извлекает с периодичностью один раз в минуту список устройств из БД устройств (Oracle) и отправляет в соответствующую очередь в кафку, которая используется для фильтрации событий.

2.6.1. Скрипт управления:

```
systemctl start|stop|status m2m_databus_devicedb_connect.service
```

2.6.2. Каталог установки:

*/opt/m2m/databus-platform/DeviceDBJDBCConnectService/*

2.6.3. Логи:

*/var/log/m2m/databus-platform/DeviceDBJDBCConnectService/*

## **2.7. AllowedDevices\_PCRF\_CDR\_Filter**

Модуль принимает поток сырых данных в формате PCRF\_CDR, а также поток списка устройств m2m (см. DeviceStorageConnector), фильтрует и пропускает только данные (в формате PCRF\_CDR), принадлежащие устройствам из списка и отправляет в результирующую очередь, которая используется модулем записи фильтрованных данных в ClickHouse (см. FilteredRawPGWCDRPusherToDB) и модулем проверки соответствия преопределенным действиям (срабатывания триггеров, см. FilteredSessionsManager).

2.7.1. Скрипт управления:

*systemctl start|stop|status m2m\_databus\_allowed\_devices\_pcrf\_cdr\_filter.service*

2.7.2. Каталог установки:

*/opt/m2m/databus-platform/AllowedDevices\_PCRF\_CDR\_Filter/*

2.7.3. Логи:

*/var/log/m2m/databus-platform/AllowedDevices\_PCRF\_CDR\_Filter/*

## **2.8. FilteredSessionsManager**

Принимает агрегированные и отфильтрованные модулем AllowedDevices\_PCRF\_CDR\_Filter PCRF\_CDR, записывает их блоками данных в таблицу БД устройств (Oracle), вызывает хранимую процедуру проверки соответствия предопределённым действиям (срабатывания триггеров), при наличии таковых получает список действий, отправка email или sms, и отправляет действия в соответствующие очереди см. EmailActionManager и SMSActionManager).

2.8.1. Скрипт управления:

*systemctl start|stop|status m2m\_databus\_filtered\_pcrf\_sessions\_manager.service*

2.8.2. Каталог установки:

*/opt/m2m/databus-platform/FilteredSessionsManager/*

2.8.3. Логи:

*/var/log/m2m/databus-platform/FilteredSessionsManager/*

## **2.9. Filtered\_PCRF\_CDR\_PusherToDB**

Принимает отфильтрованные модулем AllowedDevices\_PCRF\_CDR\_Filter данные в формате PCRF\_CDR и записывает их в БД ClickHouse

2.9.1. Скрипт управления:

*systemctl start|stop|status m2m\_databus\_filtered\_pcrf\_cdr\_pusher\_to\_db.service*

2.9.2. Каталог установки:

*/opt/m2m/databus-platform/FilteredPCRF\_CDR\_PusherToDB/*

2.9.3. Логи:

*/var/log/m2m/databus-platform/FilteredPCRF\_CDR\_PusherToDB/*

## **2.10. AllowedDevices\_SMS\_CDR\_Filter**

Модуль принимает поток сырых данных в формате SMS\_CDR, а также поток списка устройств m2m (см. DeviceStorageConnector), фильтрует и пропускает только данные (в формате SMS\_CDR), принадлежащие устройствам из списка и отправляет в результирующую очередь, которая используется модулем записи фильтрованных данных в ClickHouse (см. Filtered\_SMS\_CDR\_PusherToDB).

2.10.1. Скрипт управления:

*systemctl start|stop|status m2m\_databus\_allowed\_devices\_sms\_cdr\_filter.service*

2.10.2. Каталог установки:  
*/opt/m2m/databus-platform/AllowedDevices\_SMS\_CDR\_Filter*  
2.10.3. Логи:  
*/var/log/m2m/databus-platform/AllowedDevices\_SMS\_CDR\_Filter/*

### **2.11. Filtered\_SMS\_CDR\_PusherToDB**

Принимает отфильтрованные модулем *AllowedDevices\_SMS\_CDR\_Filter* данные в формате *SMS\_CDR* и записывает их в БД ClickHouse

2.11.1. Скрипт управления:  
*systemctl start|stop|status m2m\_databus\_filtered\_sms\_cdr\_pusher\_to\_db.service*  
2.11.2. Каталог установки:  
*/opt/m2m/databus-platform/Filtered\_SMS\_CDR\_PusherToDB/*  
2.11.3. Логи:  
*/var/log/m2m/databus-platform/Filtered\_SMS\_CDR\_PusherToDB/*

### **2.12. EmailActionManager**

Модуль читает очередь действий с сообщениями для отправки Email и отправляет соответствующие запросы в SMTP microservice.

2.12.1. Скрипт управления:  
*systemctl start|stop|status m2m\_databus\_email\_action\_manager.service*  
2.12.2. Каталог установки:  
*/opt /opt/m2m/databus-platform/EmailActionManager/*  
2.12.3. Логи:  
*/var/log/m2m/databus-platform/EmailActionManager /*

### **2.13. SMSActionManager**

Модуль читает очередь действий с сообщениями для отправки SMS и отправляет соответствующие запросы в SMPP microservice.

2.13.1. 2.12.1. Скрипт управления:  
*systemctl start|stop|status m2m\_databus\_smpp\_action\_manager.service*  
2.13.2. Каталог установки:  
*/opt/m2m/databus-platform/SMPPActionManager/*  
2.13.3. Логи:  
*/var/log/m2m/databus-platform/SMPPActionManager/*

## **3. ClickHouse**

3.1. Скрипт управления  
*systemctl start|stop|status clickhouse-server*  
3.2. Каталог инсталляции  
*/var/lib/clickhouse*  
3.3. Логи  
*/var/log/clickhouse-server*