



ЦИТАДЕЛЬ

ООО «Цитадель»
127015, г. Москва, ул. Новодмитровская, д. 2Б
+7 (495) 666 2 333, info@ctdl.ru

ПО ТС ОРМ «ОЛИМП-MVNO»

Руководство по установке и настройке

Содержание

1	Общая информация о ПО ТС ОРМ «ОЛИМП-MVNO»	3
2	Аппаратная часть ПАК «ОЛИМП-MVNO»	5
2.1	Технические требования.....	5
3	Программное обеспечение ТС ОРМ «ОЛИМП-MVNO»	6
3.1	ПО модуля «Устройство съема» (без резервирования).....	6
3.2	ПО модуля «Центральный узел» (без резервирования)	6
3.3	ПО модуля «Устройство съема» / «Центральный узел» (с резервированием)	6
3.4	ПО модуля «Шлюз ПУ»	7
4	Программные компоненты ПО ТС ОРМ «ОЛИМП-MVNO»	8
4.1	Компонент mass.....	8
4.1.1	Назначение и область применения.....	8
4.1.2	Функциональность	8
4.2	Компонент ims_fancy_db	9
4.2.1	Назначение и область применения.....	9
4.3	Компонент ims_fancy_import	9
4.3.1	Назначение и область применения.....	9
4.4	Компонент voltegw	9
4.4.1	Назначение и область применения.....	9
4.4.2	Системные требования	9
4.5	Компонент installer.....	10
4.5.1	Назначение и область применения.....	10
4.5.2	Описание работы с компонентом. Запуск	10
4.5.3	Описание аргументов	10
4.5.4	Примеры инсталляции (установка, обновление, удаление).....	10
4.5.5	Структура инсталляции	11
4.5.6	Примеры управления модулем	11
4.5.7	Установка автозапуска	12
5	Установка ПО ТС ОРМ «ОЛИМП-MVNO»	13
5.1	Установка компонентов mass, voltegw, ims_fancy_db,ims_fancy_import.....	13
5.1.1	Развертывание ПО ТС ОРМ «ОЛИМП-MVNO»	13
6	Настройка ПО ТС ОРМ «ОЛИМП-MVNO»	14
6.1	Программный компонент mass	14
6.1.1	Настройка программного компонента mass	14
6.2	Программный компонент voltegw	15
6.2.1	Настройка программного компонента voltegw	15
6.3	Программный компонент ims_fancy_db	15
6.3.1	Настройка программного компонента ims_fancy_db	15
6.4	Программный компонент ims_fancy_import.....	15
6.4.1	Настройка программного компонента ims_fancy_import.....	15
7	Обновление ПО ТС ОРМ «ОЛИМП-MVNO»	16
7.1	Обновление компонента mass	16
7.2	Обновление компонента voltegw	16
7.3	Обновление компонента ims_fancy_db	16
7.4	Обновление компонента ims_fancy_import	16
8	Работа с программными компонентами ПО ТС ОРМ «ОЛИМП-MVNO»	17

1 Общая информация о ПО ТС ОРМ «ОЛИМП-MVNO»

Программное обеспечение ТС ОРМ «ОЛИМП-MVNO» (далее – ПО ТС ОРМ «ОЛИМП-MVNO») в составе программно-аппаратного комплекса «ОЛИМП-MVNO» (далее - ПАК «ОЛИМП-MVNO») предназначено для обеспечения законного перехвата информации на сетях MVNO операторов в соответствии с требованиями Приказа Минкомсвязи России №645 от 12.12.2016 (далее – Приказ №645). Комплекс обладает возможностями расширения и модернизации с учетом развития технологий и меняющихся требований правоохранительных органов.

ПАК «ОЛИМП-MVNO» предназначено для реализации требований Приказа №645 на сетях Операторов связи, использующим бизнес-модель виртуальных сетей подвижной радиотелефонной связи стандарта GSM/UMTS/LTE, при оказании голосовых услуг.

ПО ТС ОРМ «ОЛИМП-MVNO» в составе ПАК ТС ОРМ «ОЛИМП-MVNO» предназначено для обеспечения законного перехвата на сетях операторов MVNO и передачу информации на стандартный ПУ ОРМ: ПО ТС ОРМ «ОЛИМП-MVNO» обеспечивает прием команд от ПУ ОРМ, их обработку и выполнение, а также обратную передачу требуемых голосовых потоков, ответных и статистических сообщений, сформированных из проанализированного трафика, на ПУ ОРМ.

Типовая схема работы ПО ТС ОРМ «ОЛИМП-MVNO» в составе ПАК «ОЛИМП-MVNO» представлена на рисунке 1.

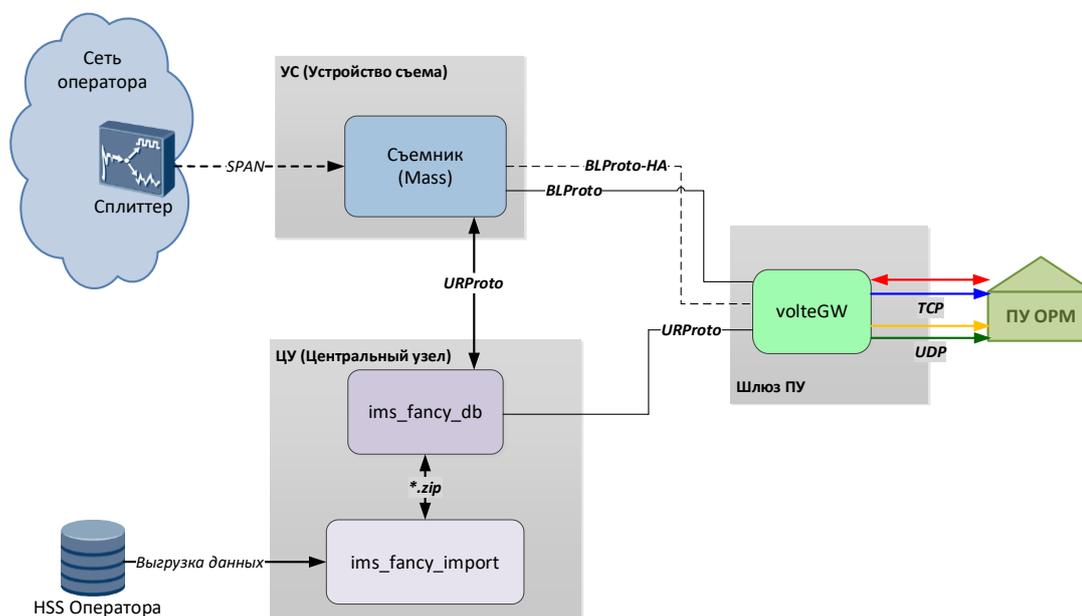


Рисунок 1. Типовая схема работы ПО ТС ОРМ «ОЛИМП-MVNO»

Также возможен вариант исполнения ПАК «ОЛИМП-MVNO» с совмещенным компонентом ЦУ/УС на одной аппаратной платформе и резервированием (рисунок 2).

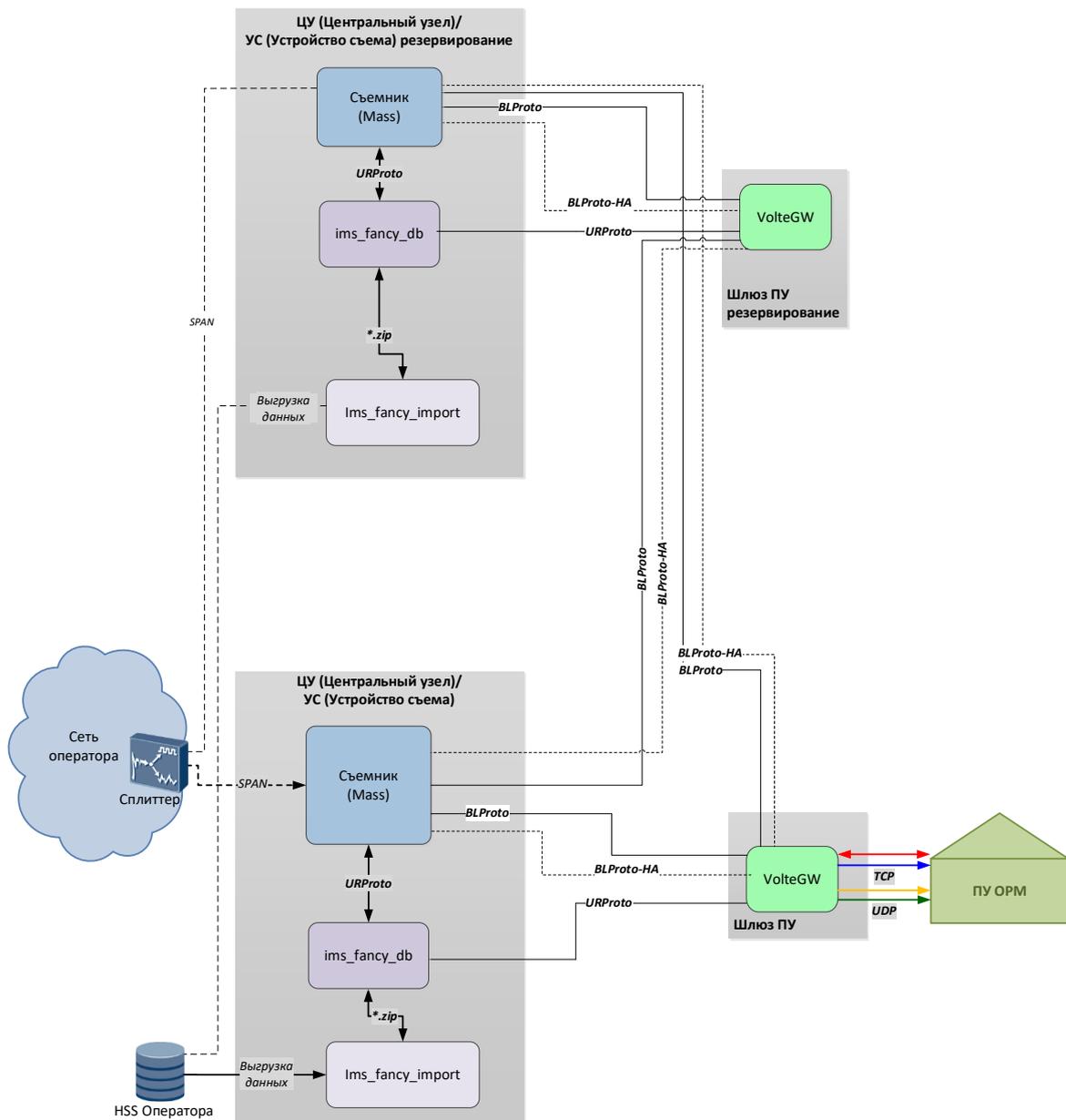


Рисунок 2. Типовая схема работы ПАК «ОЛИМП-MVNO» с резервированием

2 Аппаратная часть ПАК «ОЛИМП-MVNO»

В состав ПАК «ОЛИМП-MVNO» (в зависимости от конфигурации) входят следующие аппаратные элементы:

- «Устройство съема» (УС);
- «Центральный узел» (ЦУ);
- «Центральный узел»/ «Устройство съема» (ЦУ/УС);
- «Шлюз ПУ»;

«Устройство съема» - это аппаратно-программный модуль, обеспечивающий пассивный съём информации, поступающей с точек съема трафика на сети оператора, ее последующую обработку, а также передачу отобранной информации на ЦУ. Выполнен на базе сервера высотой 2U (6028r-wtr) и 3U (6038R-TXR).

«Центральный узел» - это программный модуль, отвечающий за создание запросов в сторону УС, а также за взаимодействие с РУ по протоколу IP для получения команд и отправки отобранной информации на РУ. Выполнен на базе сервера высотой 2U.

«Шлюз ПУ» - это аппаратно-программный модуль, предназначенный для развертывания следующего компонента:

- «Шлюз ПУ»;

2.1 Технические требования

Технические требования к аппаратной части ПО ТС ОРМ «ОЛИМП-MVNO» в составе ПАК «ОЛИМП-MVNO» приведены в таблице Таблица 1.

Таблица 1. Технические требования к аппаратной части комплекса

	Устройство съема/ Центральный узел	Шлюз	GIS
Серверная платформа	1U/2U	1U	1U
Процессор	2xIntel Xeon Gold 5120 scalable (14 core, 2,20 GHz, 19,25 MB L3)/ 2xIntel Xeon E5-2687W v4 (30M Cache, 3.00 GHz, 12 core)	2xIntel Xeon Gold 5120 scalable (14 core, 2,20 GHz, 19,25 MB L3)	1xIntel Xeon-SC 3106 (8 core, 1,70 GHz, 11 MB L3)
Оперативная память	2xDDR4 32GB (64 Гб) / 4xDDR4 16GB (64 Гб)	2xDDR4 32GB (64 Гб)	2xDDR4 16GB (32 Гб)
Операционная система	Debian	Debian	Debian
Сетевая плата	1xPE310G4SPI9LA-XR 1xIntel I350T4V2BLK /	1xPE310G4SPI9LA-XR 1x Quasar-4PCX /	-
Кол-во и тип интерфейсов (внешние)	4x1000Base-X/10GBase-X 4x1000Base-T	4x1000Base-X/10GBase-X 4xE1	-
Кол-во и тип интерфейсов (интегрированные)	1xIPMI 2x1000Base-T	1xIPMI 2x1000Base-T	1xIPMI 2x1000Base-T
Объем жесткого диска	2xHGST SATA-III 2Тб (4 Тб) / 4xHGST SATA-III 2Тб (8 Тб)	1xHDD SATA-III 1Тб (1 Тб)	1xHDD SATA-III 1Тб (1 Тб)

3 Программное обеспечение ТС ОРМ «ОЛИМП-MVNO»

3.1 ПО модуля «Устройство съема» (без резервирования)

На сервере «Устройство съема» размещаются следующие программные компоненты, отвечающие за съём, декодирование и обработку трафика:

- Библиотека захвата, отвечает за съём и отправку сетевого трафика, полученного от Оператора связи, на приложения ПАК «ОЛИМП-MVNO» без потери и искажения;
- Компонент **mass** (съёмник) отвечает за съём трафика с устройства захвата пакетов (библиотеки захвата), его запись в базу данных и отправку на модуль «Шлюз ПУ» (компонент **voltegw**) звонковых событий, голосовых пакетов, передачу событий об изменении статуса регистрации и местоположения абонента в пакетной сети.

3.2 ПО модуля «Центральный узел» (без резервирования)

На сервере Центрального Узла устанавливаются следующие программные компоненты, отвечающие за выгрузку и хранение информации:

- Программный компонент **ims_fancy_import** – отвечает за выгрузку уполномоченным органам данных по абонентам, использующим услуги;
- База данных **ims_fancy_db** – отвечает за хранение необходимой информации по абонентам, такой как местоположение, регистрация/дерегистрация, список услуг ДВО.

3.3 ПО модуля «Устройство съема» / «Центральный узел» (с резервированием)

На серверах «Устройство съема» и «Центральный узел» развернуто одинаковое количество программных компонентов.

На сервере «Устройство съема» размещаются следующие программные компоненты, отвечающие за съём, декодирование и обработку трафика, а также отвечающие за выгрузку и хранение информации:

- Библиотека захвата, отвечает за съём и отправку сетевого трафика, полученного от Оператора связи, на приложения ПАК «ОЛИМП-MVNO» без потери и искажения.
- Компонент **mass** (съёмник) отвечает за съём трафика с устройства захвата пакетов (библиотеки захвата), декодирование и обработку сетевого трафика, его запись в базу данных. В режиме активного состояния так же осуществляет отправку на модуль «Шлюз ПУ» (компонент **voltegw**) звонковых событий, голосовых пакетов, передачу событий об изменении статуса регистрации и местоположения абонента в пакетной сети;
- База данных **ims_fancy_db** – отвечает за хранение необходимой информации по абонентам, такой как местоположение, регистрация/дерегистрация, список услуг ДВО;

3.4 ПО модуля «Шлюз ПУ»

На сервере «Шлюз ПУ» размещаются следующие программные компоненты:

- Компонент **voltegw** отвечает за взаимодействие с ПУ ОРМ, формирует команды на подсистемы комплекса, получает информацию с подсистем комплекса, объединяет информацию с подсистем. Компонент **voltegw** состоит из двух частей: Сорм-шлюз и библиотека `libssrv`, реализующая требования Приказа №645.

4 Программные компоненты ПО ТС ОРМ «ОЛИМП-MVNO»

Общая схема взаимодействия программных компонентов ПО ТС ОРМ «ОЛИМП-MVNO» представлена на рисунке 3.

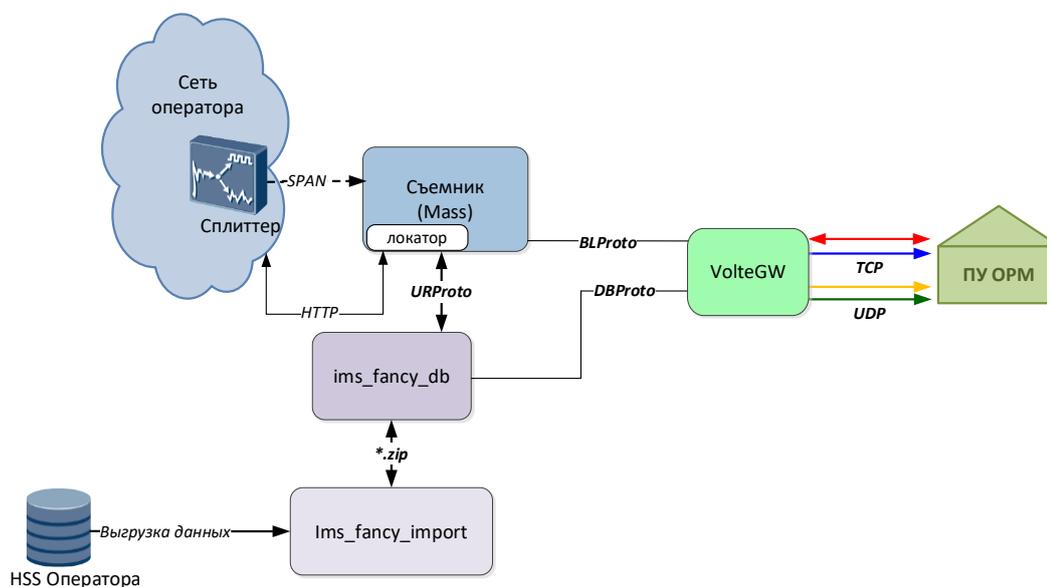


Рисунок 3. Общая схема взаимодействия программных компонентов ПО ТС ОРМ «ОЛИМП-MVNO»

4.1 Компонент mass

4.1.1 Назначение и область применения

Приложение **mass** осуществляет съём, обработку и декодирование сетевого трафика. Для съёма трафика **mass** использует библиотеку захвата трафика, а для взаимодействия с **ims_fancy_db** – библиотеку **libims_fancy_clients**.

4.1.2 Функциональность

Модуль позволяет:

- работать с входящим трафиком в режиме реального времени (с помощью библиотек захвата трафика);
- фильтровать входящий трафик по ряду параметров, задаваемых в конфигурационном файле;
- разбирать пакеты протоколов CAMEL MAP DIAMETR SIP SCTP;
- анализировать и хранить состояние абонентских сессий (создание, изменение, удаление);
- выгружать данные об изменении местоположения абонента и/или изменении состояния абонентской сессии в виде событий;
- передавать события про проприетарному протоколу получателю данных COPM IMS.

4.2 Компонент `ims_fancy_db`

4.2.1 Назначение и область применения

Компонент `ims_fancy_db` является базой данных, используется в составе ПО ТС ОРМ «ОЛИМП-MVNO» для накопления и хранения информации об абонентах и абонентских устройствах, пользующихся сервисом Оператора, контролируемым комплексом. Отличается высокой скоростью работы и простотой доступа к ней.

4.3 Компонент `ims_fancy_import`

4.3.1 Назначение и область применения

Приложение `ims_fancy_import` является парсером выгрузок, поступающих от HSS оператора связи.

4.4 Компонент `voltegw`

4.4.1 Назначение и область применения

Приложение `st_voltegw` выполняет следующие задачи:

- реализует протокол приказа для взаимодействия с ПУ ОРМ в соответствии с техническими требованиями;
- осуществляет непосредственное взаимодействие с ПУ ОРМ;
- осуществляет взаимодействие со съёмником (**mass**), в том числе в процессе резервирования;
- преобразует данные, полученные со съёмника, к формату, соответствующему техническим требованиям, для передачи на ПУ ОРМ:
 - дополняет данные информацией об абонентах, полученной от оператора связи (**ims_fancy_db**);
 - фильтрует дублирующуюся информацию;
 - генерирует недостающие события в поступающих данных (звуковой сигнал контроля посылы вызова и т.п.);
- хранит в памяти список параметров контроля, полученных с ПУ ОРМ, и передаёт их на съёмник в случае потери и восстановления связи со съёмником (например, при его перезапуске);
- формирует корректные команды постановки на контроль (т.е. вместо PID, полученного с ПУ ОРМ, будет использоваться MSISDN, UID и ISI, фигурирующие в трафике).

4.4.2 Системные требования

Требования к ОС: Debian для 64-разрядных платформ.

4.5 Компонент `installer`

4.5.1 Назначение и область применения

Компонент **Installer** – это установщик компонентов. Основное преимущество: возможность установки N независимых инстансов на сервер.

4.5.2 Описание работы с компонентом. Запуск

Внимание! Название инстансов и компонентов является примером и не является обязательным.

Запуск осуществляется исполняемым файлом находящимся в каталоге установщика `./installer` и передачей ему аргументов.

4.5.3 Описание аргументов

Аргумент `help/h`

Используется для вывода справки. Пример использования:

```
./installer -h
./installer --help
```

Также возможно использование после аргумента **in/un/up**.

Аргумент `install/in`

Используется для установки компонента и всех зависимостей в системе. Пример использования:

```
./installer in
./installer install
```

Аргумент `uninstall/un`

Используется для удаления компонента и всех зависимостей в системе.

```
./installer un
./installer uninstall
```

Аргумент `update/up`

Используется для обновления компонента и всех зависимостей в системе.

```
./installer up
./installer update
```

4.5.4 Примеры инсталляции (установка, обновление, удаление)

Установка:

```
./installer in /home/mass-x.x-x.x.zip /home/sbtk/ mvno_mass
./installer install /home/mass-x.x-x.x.zip /home/sbtk/ mvno_mass
```

Удаление:

```
./installer un /home/sbtk/ mvno_mass
./installer uninstall /home/sbtk/ mvno_mass
```

Обновление:

```
./installer up /home/mass-x.x-x.x.zip /home/sbtk/ mvno_mass
./installer update /home/mass-x.x-x.x.zip /home/sbtk/ mvno_mass
```

4.5.5 Структура инсталляции

Все компоненты **Installer** по умолчанию расположены в каталоге **/home**. Рекомендуется создавать директории по следующему правилу:

```
/
|- home
  |- product_name
    |- module_name
      |- instance_name_1
        |- bin
        |- cfg
        |- lib
        |- temp
      ...
      |- instance_name_N
        |- bin
        |- cfg
        |- lib
        |- temp
```

Директории **bin/cfg/lib/temp** создаются инсталлятором автоматически.

4.5.6 Примеры управления модулем

Запуск модуля осуществляется командой **start**, пример ниже:

```
systemctl start mass_mvno_mass.service
```

Остановка модуля осуществляется командой **stop**, пример ниже:

```
systemctl stop mass_mvno_mass.service
```

Перезапуск модуля осуществляется командой **stop/start**, пример ниже:

```
systemctl stop mass_mvno_mass.service
systemctl start mass_mvno_mass.service
```

Запрос версии осуществляется командой **version**, пример ниже:

```
mass_mvno_mass version
```

Запрос статуса осуществляется командой **status**, пример ниже:

```
mass_mvno_mass status
```

4.5.7 Установка автозапуска

Указание опции автозапуска и удаление программы из списка автозапуска осуществляется с помощью утилиты **systemctl**:

```
# Добавить экземпляр список автозапуска
sudo systemctl enable <имя_экземпляра>.service

# Удалить экземпляр из списка автозапуска
sudo systemctl disable <имя_экземпляра>.service

# Проверить статус автозапуска экземпляра
sudo systemctl is-enabled <имя_экземпляра>.service

# Просмотреть список всех имеющихся сервисов на машине
sudo systemctl list-unit-files
```

5 Установка ПО ТС ОРМ «ОЛИМП-MVNO»

5.1 Установка компонентов **mass, voltegw, ims_fancy_db,ims_fancy_import**

5.1.1 Развертывание ПО ТС ОРМ «ОЛИМП-MVNO»

Дистрибутив представляет собой папку вида **X.XX** (где X.XX - номер релиза), содержащую архивы дистрибутивов для их установки или обновления.

Папка с дистрибутивами копируется на сервер. Сначала распаковывается дистрибутив компонента **installer** в заранее созданный каталог.

Пример команды распаковки:

```
tar xaf installer-x.x-x.x.tar.gz
```

Далее производится установка компонентов комплекса, примеры описаны выше в пункте 4.5.4 описания компонента **installer**.

Структура инсталляции и примеры управления модулями так же описаны выше в пунктах 4.5.5 и 4.5.6 соответственно.

Количество устанавливаемых и запускаемых **voltegw** определяется количеством подключаемых ПУ ОРМ.

6 Настройка ПО ТС ОРМ «ОЛИМП-MVNO»

6.1 Программный компонент mass

6.1.1 Настройка программного компонента mass

Для настройки программного модуля **mass**:

- 1) настройте файл конфигурации **mass.ini** ;
- 2) настройте лог-файл **mass.log4cplus** ;
- 3) в системном файрволе (**iptables**) разрешите входящие подключения к порту, указанному в секции [**voltegw**] и исходящие подключения на порт, указанный в секции [**repo**].

6.1.1.1 Файл конфигурации

Для настройки приложения используются файлы конфигурации, которые располагаются в каталоге, где установлен модуль **mass**. Файлы доступны для редактирования в любом текстовом редакторе. Перед редактированием необходимо скопировать конфигурационный файл во временный и производить все изменения в нем, а по окончании редактирования записать временный файл вместо первоначального.

Внимание! После внесения изменений в конфигурационный файл требуется перезапуск приложения.

Внимание!

1) Все протоколы верхнего и нижнего уровня, в том числе транспортные протоколы – TCP, UDP, SCTP – включены по умолчанию.

Включаемые протоколы:

- **SIP** – (Session Initiation Protocol) протокол установления сеанса;
- **GTP** – (GPRS Tunnelling Protocol) протокол туннелирования GPRS;
- **ESP** – (Encapsulating Security Payload) протокол защиты (шифрования) данных;
- **TLS** – (Transport Layer Security) протокол защиты транспортного уровня;
- **MSRP** – (Message Session Relay Protocol) протокол передачи последовательности связанных мгновенных сообщений в рамках сеанса связи;
- **HTTP** – (HyperText Transfer Protocol) — протокол передачи гипертекста;
- **GRE** – (Generic Routing Encapsulation) протокол туннелирования сетевых пакетов;
- **DIAMETER** – сеансовый протокол, обеспечивающий взаимодействие между клиентами в целях аутентификации, авторизации и учёта различных сервисов (AAA);
- **MAP** – (Mobile Application Part - прикладная подсистема подвижной связи в сетях стандартов GSM и UMTS) протокол прикладного уровня, предназначенный для доступа к HLR, VLR, MSC, EIR, AAA, SMSC, SGSN;
- **CAMEL** – (CAMEL Application Part (CAP)) протокол прикладного уровня, реализующий интеллектуальные услуги внутри сетей GSM и UMTS;

- **MEF8** – эмуляция TDM-цепей PDH для передачи по сетям Metro-Ethernet.

6.2 Программный компонент **voltegw**

6.2.1 Настройка программного компонента **voltegw**

6.2.1.1 Общий порядок действий

Для настройки модуля необходимо выполнить следующие шаги.

- 1) Настройте конфигурационный файл модуля **voltegw.ini**, соблюдая следующий порядок:
 - настройка параметров соединения с ПУ (приказ, порты, адреса, протоколы, кодеки для звука и т.д.);
 - настройка соединения со съёмником или съёмниками (адреса, порты);
 - настройка соединения с БД абонентов (порты, адреса, версия базы);
 - общие настройки (оператор, сервис оператора, правила кодирования местоположения и т.д.);
 - прочие опциональные настройки.
- 2) Настройте конфигурационный файл логирования **log4d.props**.
- 3) В системном файрволе (iptables и прочее) разрешите входящие и исходящие подключения для портов, заданных в конфигурационных файлах.

6.2.1.2 Описание конфигурационных файлов модуля

В компоненте **voltegw** используются конфигурационные файлы в кодировке UTF-8 (без BOM). Файлы доступны для редактирования в любом текстовом редакторе, обеспечивающем запись в простом текстовом формате. Перед редактированием необходимо скопировать конфигурационный файл во временный и производить все изменения в нём, а по окончании редактирования записать временный файл вместо первоначального.

После внесения изменений в конфигурационные файлы требуется перезапуск модуля.

6.3 Программный компонент **ims_fancy_db**

6.3.1 Настройка программного компонента **ims_fancy_db**

Файлы конфигурации **instances.yml** и **tarantool.yml** расположены в каталоге с установленным модулем **ims_fancy_db**. Пример: `/home/sbtk/ims_db/cfg`

6.4 Программный компонент **ims_fancy_import**

6.4.1 Настройка программного компонента **ims_fancy_import**

Файл конфигурации **ims_fancy_import.ini** расположен в каталоге с установленным модулем **ims_fancy_import**. Пример: `/home/sbtk/ims_import/cfg`

7 Обновление ПО ТС ОРМ «ОЛИМП-MVNO»

7.1 Обновление компонента **mass**

Для обновления компонента **mass** необходимо произвести обновление с помощью модуля **installer** и полученного дистрибутива модуля **mass**. Пример приведен выше в пункте 4.5.4 описания компонента **installer**.

7.2 Обновление компонента **voltegw**

Для обновления компонента **voltegw** необходимо произвести обновление с помощью модуля **installer** и полученного дистрибутива модуля **voltegw**. Пример приведен выше в пункте 4.5.4 описания компонента **installer**.

7.3 Обновление компонента **ims_fancy_db**

Для обновления компонента **ims_fancy_db** необходимо произвести обновление с помощью модуля **installer** и полученного дистрибутива модуля **ims_fancy_db**. Пример приведен выше в пункте 4.5.4 описания компонента **installer**.

7.4 Обновление компонента **ims_fancy_import**

Для обновления компонента **ims_fancy_import** необходимо произвести обновление с помощью модуля **installer** и полученного дистрибутива модуля **ims_fancy_import**. Пример приведен выше в пункте 4.5.4 описания компонента **installer**.

8 Работа с программными компонентами ПО ТС ОРМ «ОЛИМП-MVNO»

Для работы с ПО «ОЛИМП-MVNO» используется служба **systemd**:

- 1) Управление каждым модулями через **systemd** после установки с помощью модуля **installer**. Примеры описаны выше в пункте 4.5.6 описания компонента **installer**.