



ЦИТАДЕЛЬ

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

ПО ТС ОРМ «ОЛИМП-УРВХ»

Руководство по эксплуатации

Содержание

1	Общая информация о ПО ТС ОРМ «ОЛИМП-VPBX»	3
2	Программное обеспечение ТС ОРМ «ОЛИМП-VPBX»	4
2.1	ПО модуля «Устройство съема».....	4
2.2	ПО модуля «Центральный узел»	4
2.3	ПО модуля «Шлюз ПУ»	4
3	Программные компоненты ПО ТС ОРМ «ОЛИМП-VPBX»	5
3.1	Компонент mass.....	5
3.1.1	Назначение и область применения.....	5
3.1.2	Функциональность	5
3.2	Компонент vrbx_gero_sgv2.....	6
3.2.1	Назначение и область применения.....	6
3.3	Компонент vrbx_import2	6
3.3.1	Назначение и область применения.....	6
3.4	Компонент voltegw	6
3.4.1	Назначение и область применения.....	6
3.4.2	Системные требования	7
3.5	Компонент installer.....	7
3.5.1	Назначение и область применения.....	7
3.5.2	Описание работы с компонентом. Запуск	7
3.5.3	Описание аргументов	7
3.5.4	Примеры инсталляции (установка, обновление, удаление).....	7
3.5.5	Структура инсталляции	8
3.5.6	Примеры управления модулем	8
3.5.7	Установка автозапуска	9
4	Работа с программными компонентами ПАК «ОЛИМП-VPBX»	10

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

Программное обеспечение ТС ОРМ «ОЛИМП-VPBX» (далее – ПО ТС ОРМ «ОЛИМП-VPBX») в составе программно-аппаратного комплекса «ОЛИМП-VPBX» (далее - ПАК «ОЛИМП-VPBX») предназначено для обеспечения законного перехвата информации на сетях VPBX 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-VPBX» 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-VPBX» в составе ПАК ТС ОРМ «ОЛИМП-VPBX» предназначено для обеспечения законного перехвата на сетях операторов и передачу информации на стандартный ПУ ОРМ: ПО ТС ОРМ «ОЛИМП-VPBX» 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-VPBX» в составе ПАК «ОЛИМП-VPBX» представлена на рисунке 1.

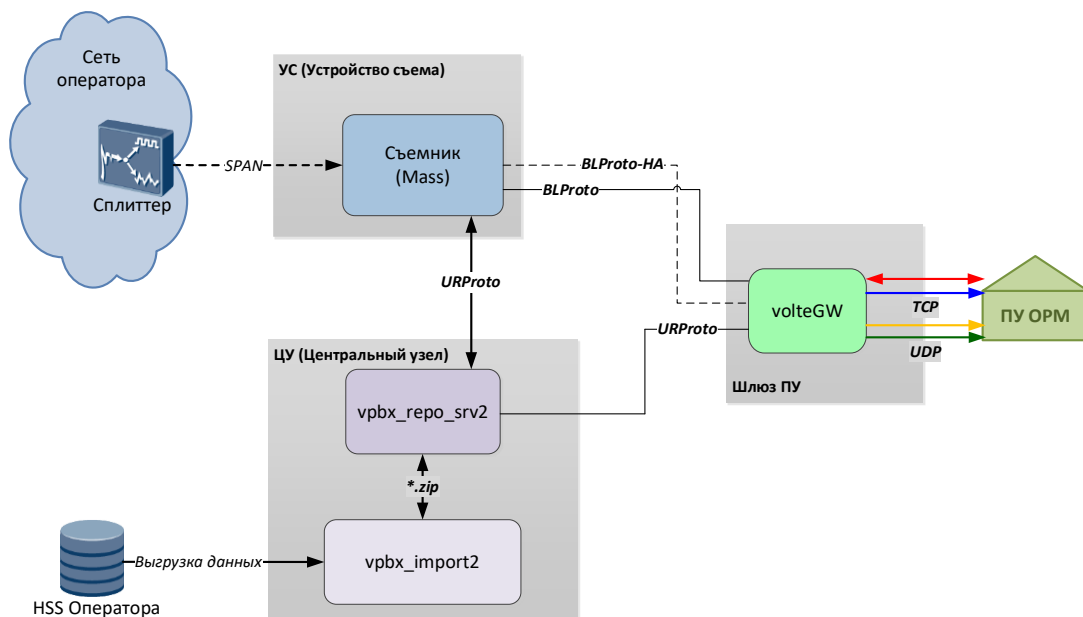


Рисунок 1. Типовая схема работы ПО ТС ОРМ «ОЛИМП-VPBX» в составе ПАК «ОЛИМП-VPBX»

2 Программное обеспечение ТС ОРМ «ОЛИМП-VPBX»

2.1 ПО модуля «Устройство съема»

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 **mass** ПАК « ОЛИМП-VPBX » 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.

2.2 ПО модуля «Центральный узел»

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

- Программный компонент **vpbx_import2**– отвечает за выгрузку уполномоченным органам данных по абонентам, использующим услуги;
- База данных **vpbx_repo_srv2**– отвечает за хранение необходимой информации по абонентам, такой как местоположение, регистрация/дерегистрация, список услуг ДВО.

2.3 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.

3 Программные компоненты ПО ТС ОРМ «ОЛИМП-VPBX»

Общая схема взаимодействия программных компонентов ПО ТС ОРМ «ОЛИМП-VPBX» представлена на рисунке 2.

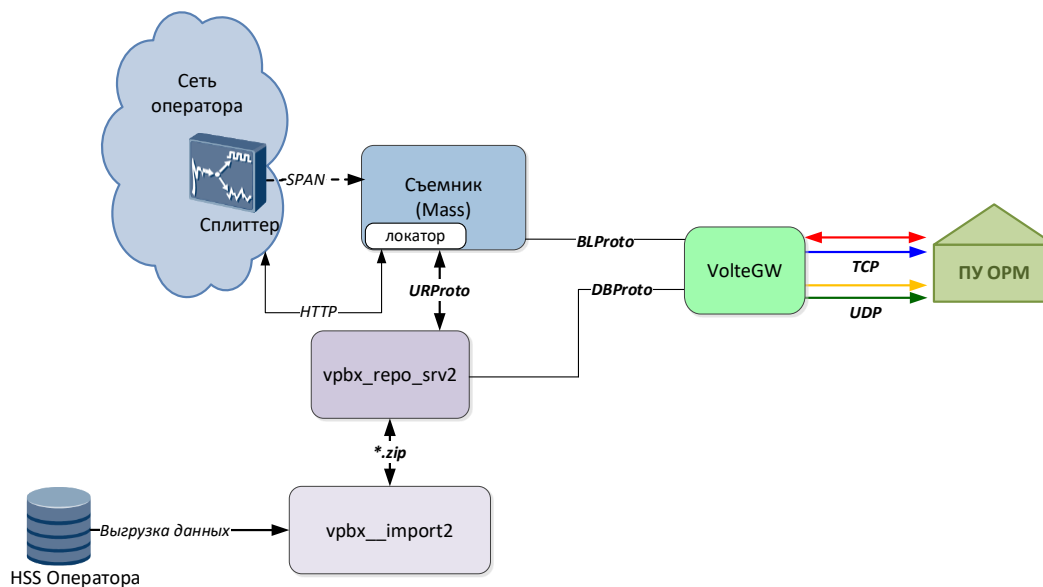


Рисунок 2. Общая схема взаимодействия программных компонентов ПАК «ОЛИМП-VPBX»

3.1 Компонент mass

3.1.1 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 **vpbx_repo_srv2** – библиотеку **libvpbxreptomassclient2**.

3.1.2 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 BCST TLS 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 по проприетарному протоколу получателю данных СОРМ VPBX.

3.2 Компонент `vpbx_repo_srv2`

3.2.1 Назначение и область применения

Компонент `vpbx_repo_srv2` является базой данных, которая используется в составе ПО ТС ОРМ «ОЛИМП-VPBX» для накопления и хранения информации об абонентах и абонентских устройствах, пользующихся сервисом Оператора, контролируемым комплексом. Отличается высокой скоростью работы и простотой доступа к нему.

3.3 Компонент `vpbx_import2`

3.3.1 Назначение и область применения

Приложение `vpbx_import2` является парсером выгрузок, поступающих от HSS оператора связи.

3.4 Компонент `voltegw`

3.4.1 Назначение и область применения

Приложение `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 (`vpbx_repo_srv2`);
 - 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).

3.4.2 Системные требования

Требования к ОС: Linux для 64-разрядных платформ с ядром 6.2.0-1

3.5 Компонент `installer`

3.5.1 Назначение и область применения

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

3.5.2 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.

3.5.3 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
```

3.5.4 Примеры инсталляции (установка, обновление, удаление)

Установка:

```
./installer in /home/mass-x.x-x.x.zip /home/vpbx/ vpbx_mass
./installer install /home/mass-x.x-x.x.zip /home/vpbx/ vpbx_mass
```

Удаление:

```
./installer un /home/vpbx/ vpbx_mass
./installer uninstall /home/vpbx/ vpbx_mass
```

Обновление:

```
./installer up /home/mass-x.x-x.x.zip /home/vpbx/ vpbx_mass
./installer update /home/mass-x.x-x.x.zip /home/vpbx/ vpbx_mass
```

3.5.5 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.

3.5.6 Примеры управления модулем

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

```
systemctl start mass_vpbx_mass.service
```

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

```
systemctl stop mass_vpbx_mass.service
```

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

```
systemctl stop mass_vpbx_mass.service
systemctl start mass_vpbx_mass.service
```

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

```
mass_vpbx_mass version
```

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


```
mass_vpbx_mass status
```

3.5.7 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
```

4 Работа с программными компонентами ПАК «ОЛИМП-VPBX»

Для работы с ПО ТС ОРМ «ОЛИМП-VPBX» используется служба **systemd**:

- 1) Управление каждым модулями через **systemd** после установки с помощью модуля **installer**. Примеры описаны выше в пункте [3.5.6](#) описания компонента **installer**.