

# **СЕРВИСНАЯ ШИНА ЕЦП СМП**

**Руководство администратора**

**Листов 25**

**Содержание**

<b>ПЕРЕЧЕНЬ СОКРАЩЕНИЙ .....</b>	<b>3</b>
<b>1 ВВЕДЕНИЕ .....</b>	<b>4</b>
1.1 Назначение Программы .....	4
1.2 Основные функции Программы .....	4
1.3 Сведения о технических и программных средствах, обеспечивающих выполнение программы.....	4
<b>2 СТРУКТУРА ПРОГРАММЫ .....</b>	<b>9</b>
<b>3 УСТАНОВКА И НАСТРОЙКА ПРОГРАММЫ.....</b>	<b>11</b>
<b>4 ПРОВЕРКА ПРОГРАММЫ.....</b>	<b>23</b>
<b>5 СООБЩЕНИЯ АДМИНИСТРАТОРУ .....</b>	<b>24</b>
5.1 Сообщения об ошибках.....	24
5.2 Протоколирование ошибок и предупреждений.....	25

## Перечень сокращений

Используемые в настоящем документе термины и сокращения представлены в Таблица 1.

Таблица 1 – Перечень терминов и сокращений

<b>Термин / Сокращение</b>	<b>Расшифровка</b>
АТС	Автоматическая телефонная станция
ЕИР	Единый информационный ресурс
ЕЦП СМП	Единая цифровая платформа скорой медицинской помощи
ИС	Информационная система
ИС СМП	Информационная система скорой медицинской помощи субъекта РФ
ОС	Операционная система
ПО	Программное обеспечение
ПТК	Программно-технический комплекс
ПЭВМ	Персональная электронно-вычислительная машина
РМИС	Региональная медицинская информационная система здравоохранения субъекта РФ
Система-112	Система обеспечения вызова экстренных оперативных служб по единому номеру «112» на территории субъекта РФ
СМП	Скорая медицинская помощь
Сервисная шина ЕЦП СМП, Программа	Программа для ЭВМ «Сервисная шина ЕЦП СМП»
ТФОМС	Территориальный фонд обязательного медицинского страхования субъекта РФ
ЭВМ	Электронно-вычислительная машина

# **1 ВВЕДЕНИЕ**

## **1.1 Назначение Программы**

Сервисная шина ЕЦП СМП предназначена для организации информационного обмена между распределенными региональными ИС и прикладным программным обеспечением, задействованными при оказании скорой медицинской помощи.

## **1.2 Основные функции Программы**

Программа обеспечивает информационное взаимодействие следующих ИС и программного обеспечения:

- ИС СМП;
- Система-112;
- РМИС;
- ЕИР ТФОМС;
- ПО сервисной платформы цифровой АТС.

Программа применяется в медицинских организациях субъекта Российской Федерации, оказывающих скорую и неотложную медицинскую помощь и использующих специализированные информационные системы и программное обеспечение.

## **1.3 Сведения о технических и программных средствах, обеспечивающих выполнение программы**

Стабильная работа Программы обеспечивается на серверном программно-техническом комплексе.

Тип оборудования серверного ПТК, его количественные и качественные характеристики выбираются исходя из количества подключенных ИС и ПО. Типовой сервер для субъекта РФ с населением до 5 млн. человек должен обладать следующими характеристиками:

- CPU – 12 ядер
- RAM – 16 ГБ
- HDD – 2x200 ГБ HDD 10k
- IOPS(R/W) – 1000.

Также должен быть предусмотрен АРМ администратора для установки и настройки Программы в следующем составе:

- ПЭВМ:
  - Процессор с тактовой частотой, не менее, 2 ГГц;
  - Оперативная память объемом, не менее, 4 ГБ;

- Жесткий диск объемом, не менее, 300 ГБ;
  - Монитор: диагональ, не менее, 21”, разрешение, не хуже 1920×1080;
  - Наличие сетевого адаптера для подключения к ЛВС 100/1000 Мбит/с.
- ПО:
- Браузер Google Chrome версии 85.

Программа функционирует на серверном оборудовании под управлением операционной системы Ubuntu Server. Ubuntu Server – свободно распространяемая по лицензии GNU GPL ОС для управления web-серверами, основанная на дистрибутиве Debian GNU/Linux.

Для расширения возможностей ОС на ЭВМ устанавливаются программные средства, перечисленные в Таблица 2.

Таблица 2 – Программные средства

№ п/п	ПО, расширяющее возможности ОС	Назначение
1	GCC	Набор компиляторов GCC для различных языков программирования разработан в рамках проекта GNU и является свободно распространяемым ПО. GCC является ключевым компонентом GNU toolchain и используется как стандартный компилятор для UNIX-подобных ОС. Информация по установке и настройке средства расположена в сети Интернет по адресу <a href="https://gcc.gnu.org">https://gcc.gnu.org</a> .
2	OpenSSL	Криптографический пакет с открытым исходным кодом OpenSSL для работы с SSL/TLS. Информация по установке и настройке средства расположена в сети Интернет по адресу <a href="https://www.openssl.org">https://www.openssl.org</a> .
3	libMemcached	Библиотека libMemcached предназначена для настройки взаимодействия с серверами Memcached и реализована на языках C и C++. Информация по установке и настройке средства расположена в сети Интернет по адресу <a href="http://libmemcached.org">http://libmemcached.org</a> .

№ п/п	ПО, расширяющее возможности ОС	Назначение
4	Supervisor	Клиент-серверная система supervisor позволяет администратору контролировать подключенные процессы в ОС типа UNIX. Информация по установке и настройке средства расположена в сети Интернет по адресу <a href="http://supervisord.org">http://supervisord.org</a> .
5	Memcached	ПО Memcached реализует сервис кэширования данных в оперативной памяти на основе хеш-таблицы. Информация по установке и настройке средства расположена в сети Интернет по адресу <a href="http://www.memcached.org">http://www.memcached.org</a> .
6	PHP 7.3.15	Язык программирования PHP поддерживается большинством хостинг-провайдеров и применяется для создания динамических веб-сайтов. Руководство по настройке средства доступно в сети Интернет по адресу <a href="https://php.net/">https://php.net/</a> .
7	Laravel 5.8	Кроссплатформенный свободно распространяемый по лицензии MIT веб-фреймворк Laravel с открытым кодом предназначен для разработки приложений с использованием архитектурной модели MVC. Исходный код проекта размещен на GitHub. Информация по установке и настройке средства расположена в сети Интернет по адресу <a href="http://laravel.com/">http://laravel.com/</a> .
8	Nginx web-server 1.13.0	Веб-сервер и почтовый прокси-сервер nginx работает на Unix-подобных операционных системах. Информация по установке и настройке средства расположена в сети Интернет по адресу <a href="https://nginx.org">https://nginx.org</a> .
9	MongoDB	Реализованная на языке C++ документоориентированная СУБД MongoDB с открытым исходным кодом не требует описания схемы таблиц и использует JSON-подобные документы и схему БД. Информация по установке и настройке средства расположена в сети Интернет по адресу <a href="https://www.mongodb.com/">https://www.mongodb.com/</a> .

№ п/п	ПО, расширяющее возможности ОС	Назначение
10	PostgreSQL	<p>Профессиональная объектно-реляционная СУБД PostgreSQL с открытыми исходными текстами является надежным, интегрированным и масштабируемым средством.</p> <p>Информация по установке и настройке средства расположена в сети Интернет по адресу <a href="https://www.postgresql.org/">https://www.postgresql.org/</a>.</p>
11	RabbitMQ	<p>Свободно распространяемая по лицензии MPL платформа RabbitMQ реализует систему обмена сообщениями между компонентами программной системы на основе стандарта AMQP.</p> <p>Информация по установке и настройке средства расположена в сети Интернет по адресу <a href="http://www.rabbitmq.com/">http://www.rabbitmq.com/</a>.</p>
12	Mapnik	<p>Свободный инструментарий отрисовки карты Mapnik используется для отрисовки основного Slippy Map слоя открытых картографических данных OSM.</p> <p>Информация по установке и настройке средства расположена в сети Интернет по адресу <a href="http://mapnik.org">http://mapnik.org</a>.</p>
13	Redis	<p>Резидентная СУБД Redis класса NoSQL с открытым исходным кодом работает со структурами данных типа «ключ – значение» используется как для БД, так и для реализации кэшей, брокеров сообщений.</p> <p>Информация по установке и настройке средства расположена в сети Интернет по адресу <a href="https://redis.io/">https://redis.io/</a>.</p>
14	HAProxy	<p>Свободно-распространяемый прокси-сервер HAProxy для ОС Linux, Solaris и FreeBSD предоставляет возможность балансировки нагрузки TCP/HTTP, предназначенный для повышения производительности серверной среды путем распределения рабочей нагрузки среди группы серверов ЦПС СМП.</p> <p>Информация по установке и настройке средства расположена в сети Интернет по адресу <a href="https://www.haproxy.org/">https://www.haproxy.org/</a>.</p>

№ п/п	ПО, расширяющее возможности ОС	Назначение
15	PgBouncer	Используется в качестве оптимизатора пула соединений для СУБД PostgreSQL 10.5 выступает ПО с открытым кодом PgBouncer. Информация по установке и настройке средства расположена в сети Интернет по адресу <a href="https://www.pgouncer.org/">https://www.pgouncer.org/</a> .
16	Minio	В качестве сервера хранения объектов ЦПС СМП используется решение Minio 2018-09-01T00:38:25Z. Информация по установке и настройке средства расположена в сети Интернет по адресу <a href="https://min.io/">https://min.io/</a> .



## 2 СТРУКТУРА ПРОГРАММЫ

Сервисная шина ЕЦП СМП представляет собой программное обеспечение, размещаемое на серверном оборудовании.

Программа функционирует автономно.

Программа состоит из сервисов, логически соответствующих интегрируемым ИС и ПО. Перечень сервисов и их назначение представлены в Таблица 3.

Таблица 3 – Сервисы Программы и их функции

№ п/п	Наименование	Назначение
1	Сервис взаимодействия с ИС СМП	Сервис предназначен для обеспечения информационного взаимодействия с ИС СМП субъекта РФ
2	Сервис взаимодействия с Система-112	Сервис предназначен для обеспечения информационного взаимодействия с системой обеспечения вызова экстренных оперативных служб по единому номеру «112» на территории субъекта РФ
3	Сервис взаимодействия с ЕИР ТФОМС	Сервис предназначен для обеспечения информационного взаимодействия с единым информационным ресурсом ТФОМС субъекта РФ
4	Сервис взаимодействия с РМИС	Сервис предназначен для обеспечения информационного взаимодействия с региональной медицинской информационной системой здравоохранения субъекта РФ
5	Сервис взаимодействия с ПО АТС	Модуль предназначен для обеспечения интеграционного взаимодействия с ПО цифровой АТС Заказчика

Все подключаемые к Сервисной шине ЕЦП СМП информационные системы и программное обеспечение используются в процессе деятельности организаций субъекта Российской Федерации, задействованных в оказании скорой и неотложной медицинской помощи (Рисунок 1).



Рисунок 1 – Схема взаимодействия Сервисной шины ЕЦП СМП

Подробное описание логической структуры СПО приведено в документе «Сервисная шина ЕЦП СМП. Описание программы».

### 3 УСТАНОВКА И НАСТРОЙКА ПРОГРАММЫ

Процесс установки и настройки ИС СМП

В качестве примера в описании установки программы далее будет использоваться имя пользователя cis и ip-адрес сервера 192.168.1.1001.

1) Авторизоваться на сервере:

```
$ ssh cis@192.168.1.100
```

2) Добавить ключи репозитория, из которых будет происходить дальнейшая установка всех зависимостей PHP7.

Скачать и установить ключ для репозитория с пакетами PHP7:

```
$ wget https://www.dotdeb.org/dotdeb.gpg
```

```
$ sudo apt-key add dotdeb.gpg
```

3) Установить служебные утилиты для работы с репозиториями:

```
$ sudo apt-get install apt-transport-https ca-certificates curl git php-pear
```

Добавить адрес для репозитория с пакетами PHP7. Для этого необходимо повысить привилегии в системе до пользователя root, выполнив команду `sudo -i` (в текущем сеансе пользователя). При этом приглашение консоли изменится со знака \$ на знак #.

**ВНИМАНИЕ!** Будьте внимательны при использовании учётной записи суперпользователя.

```
# echo -e "deb http://packages.dotdeb.org jessie all\ndeb-src http://packages.dotdeb.org jessie all" > /etc/apt/sources.list.d/dotdeb.list
```

4) После добавления адреса вернуться к прежнему пользователю с помощью команды `exit`. Прежде чем продолжить установку необходимо обновить список доступных репозитория, выполнив команду:

```
$ sudo apt-get update
```

5) Установить требуемые пакеты PHP7:

```
$ sudo apt-get install php7.3-fpm php7.3-pgsql php7.3-gd php7.3-soap php7.3-memcached php7.3-mongodb php7.3-mcrypt php7.3-mbstring php7.3-bcmath php7.3-dev php7.3-curl php7.3-zip php7.3-xml
```

6) Установить необходимые библиотеки для сборки ZeroMQ:

```
$ sudo apt-get install libzmq-dev pkg-config
```

```
$ sudo pecl install zmq-beta
```

7) Ввиду того, что не все пакеты PHP7 доступны для установки из репозитория, необходимо собрать их из исходных кодов.

Скачать архив ZeroMQ из репозитория на Github:

```
$ wget https://github.com/zeromq/libzmq/releases/download/v4.2.0/zeromq-4.2.0.tar.gz
```

Распаковать скачанный архив с исходными кодами:

```
$ tar -xvf zeromq-4.2.0.tar.gz
```

Перейти в директорию с исходными кодами, которые были извлечены из архива:

```
$ cd zeromq-4.2.0
```

Запустить генерацию установочной конфигурации:

```
$ ./autogen.sh
```

Выполнить конфигурирование и проверку сборки:

```
$ ./configure && make check
```

Проверка сборки должна завершиться без ошибок иначе дальнейшая установка не будет выполнена корректно.

Выполнить сборку пакета ZeroMQ:

```
$ sudo make install
```

После завершения сборки необходимо выполнить динамическое связывание ссылок для библиотеки ZeroMQ:

```
$ sudo ldconfig
```

```
$ cd ../
```

8) Установить сервер баз данных, менеджер задач, веб-сервер и т.д. (настройка будет произведена позднее):

```
$ sudo apt-get install postgresql postgres supervisor nginx memcached libmemcached-tools
```

9) Установить wkhtmltopdf (библиотека для генерации pdf-документов в браузерах на основе WebKit):

```
$ sudo apt-get install wkhtmltopdf
```

10) Установить LibreOffice и вспомогательные библиотеки для корректной работы (отображения) печатных форм:

```
$ sudo apt-get install libreoffice libxinerama-dev libdbus-glib-1-dev libcups2-dev libgl1-mesa-glx libsm6
```

11) Установить MongoDB для хранения NoSQL-данных о действиях пользователей в системе. Для этого добавить ключ репозитория:

```
$ sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv EA312927
```

Добавить адрес для репозитория с пакетом MongoDB. Для этого необходимо повысить привилегии в системе до пользователя root, выполнив команду `sudo -i` (в текущем сеансе пользователя). При этом приглашение консоли изменится со знака \$ на знак #.

**ВНИМАНИЕ!** Будьте внимательны при использовании учётной записи суперпользователя.

```
# echo "deb http://repo.mongodb.org/apt/ubuntu xenial/mongodb-org/4.4 multiverse" | tee /etc/apt/sources.list.d/mongodb-org-4.4.list
```

После добавления адреса вернуться к прежнему пользователю с помощью команды `exit`. Для продолжения установки необходимо обновить список доступных репозиториях, командой:

```
$ sudo apt-get update
```

Установить MongoDB (Community Edition):

```
$ sudo apt-get install -y mongodb-org
```

12) Перейти к настройке MongoDB. Сделать резервную копию конфигурационного файла (с настройками по умолчанию).

```
$ sudo cp /lib/systemd/system/mongod.service /lib/systemd/system/mongod.service.bak
```

Открыть конфигурационный файл, который был скопирован `/lib/systemd/system/mongod.service`:

```
$ sudo nano /lib/systemd/system/mongod.service
```

Изменить его содержимое следующей конфигурацией:

```
[Unit]
```

```
Description=High-performance, schema-free document-oriented database
```

```
After=network.target
```

```
Documentation=https://docs.mongodb.org/manual
```

```
[Service]
```

```
User=mongodb
```

```
Group=mongodb
```

```
ExecStart=/usr/bin/mongod --quiet --config /etc/mongod.conf
```

```
[Install]
```

```
WantedBy=multi-user.target
```

Сохранить конфигурационный файл, и перезагрузить сервер MongoDB:

```
$ sudo service mongod restart
```

13) Настроить веб-сервер и его окружения. В директории `/etc/nginx/sites-enable/` заменить содержимое файла `default` на представленное ниже:

```
map $http_upgrade $connection_upgrade {
```

```
    default upgrade;
```

```
    " close;
```

```
}
```

```
upstream websocket {
```

```
    server 127.0.0.1:9090;
```

```
}
```

```
server {
```

```
    proxy_max_temp_file_size 0;
```

```
listen          80;
sendfile        on;
tcp_nopush     on;
tcp_nodelay    on;

keepalive_timeout 65;
    root /data/www/http/public;
    access_log /data/www/logs/nginx_access.log;
    error_log /data/www/logs/nginx_error.log;
location / {
    try_files $uri $uri/ /index.php$is_args$args;
}
location /status {
    stub_status;
}
location ~ ^/(fpmstatus|ping)$ {
    access_log off;
    allow all;
    allow 127.0.0.1;
    allow 172.16.0.0/24;
    deny all;
    include fastcgi_params;
    fastcgi_index index.php;
    fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
    fastcgi_pass 127.0.0.1:9000;
}
location /pusher/ {
    proxy_pass http://websocket;
    proxy_http_version 1.1;
    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header Connection $connection_upgrade;
    proxy_pass_request_headers on;
}
client_max_body_size 256m;
index index.html index.php;

location ~ \.php$ {
    fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
    fastcgi_pass 127.0.0.1:9000;
```

```
    fastcgi_index index.php;
    fastcgi_split_path_info ^(.+\.(php|.*))$.
    include fastcgi_params;
    fastcgi_buffers 32 32k;
    fastcgi_buffer_size 32k;
    try_files $uri =404;
}
location ~ /\. {
    deny all;
}
location ~*\.(css|gif|ico|jpeg|jpg|js|svg|png)$ {
    expires 1h;
}
}

server {
    listen [::]:444 ipv6only=off;

    location / {
        proxy_pass http://websocket;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection $connection_upgrade;
        proxy_pass_request_headers on;
        proxy_read_timeout 65535;
    }
}
```

Данный конфигурационный файл подразумевает, что корневая директория веб-сервера хранится в каталоге `/data/www/http/public`.

**ВНИМАНИЕ!** Будьте внимательны при настройке (должна быть создана указанная директория, и её владельцем должны быть пользователь `www-data` и группа `www-data`).

Настроить РНР-интерпретатор, который будет «общаться» с веб-сервером. Для настройки `php7.3-fpm` необходимо скопировать представленный ниже конфигурационный файл в директорию `/etc/php/7.3/fpm/pool.d/` и переименовать его `www.conf`.

```
[www]
user = www-data
group = www-data
```

```
listen = 0.0.0.0:9000
listen.owner = www-data
pm = ondemand
pm.max_children = 124
pm.start_servers = 32
pm.min_spare_servers = 2
pm.max_spare_servers = 4
pm.max_requests = 1024
pm.status_path = /fpmstatus
listen.backlog = -1
php_admin_value[short_open_tag] = 1
php_admin_value[memory_limit] = 1024M
request_terminate_timeout = 300s
php_value[max_execution_time] = 120
php_value[post_max_size] = 256M
php_value[upload_max_filesize] = 256M
```

```
php_admin_value[error_log] = /data/www/logs/fpm-php.www.log
php_admin_flag[log_errors] = on
catch_workers_output = yes
```

- 14) Создать иерархию директорий для корневой директории веб-сервера и логов:

```
$ sudo mkdir /data
$ sudo mkdir /data/www
$ sudo mkdir /data/www/http
$ sudo mkdir /data/www/logs
```

- 15) Скопировать архив с исходными кодами, предоставленными компанией-разработчиком (ООО «КИ Системы»), и распаковать на уровень выше корневой директории веб-сервера /data/www/http

```
$ sudo cp ssmg.gz /data/www/http/
$ cd /data/www/http
$ sudo gzip -d /data/www/http/ssmg.gz
$ sudo chown -R www-data:www-data /data/www/
$ sudo chmod -R 777 /data/www/
```

- 16) Перезапустить nginx и php7.3-fpm, которые были сконфигурированы, чтобы вступили в силу все внесённые изменения.

```
$ sudo service php7.3-fpm restart
$ sudo service nginx restart
```



17) Настроить сервер баз данных PostgreSQL. Открыть файл конфигурации, отвечающий за разграничение доступа `pg_hba.conf`, находящийся в директории `/etc/postgresql/11/main/`, и добавить ниже приведённые строки:

```
# TYPE DATABASE USER ADDRESS METHOD
local all postgres trust
local all all md5
# IPv4
host all all 127.0.0.1/24 md5
host all all 192.168.1.0/24 md5
host all all 10.50.71.0/24 md5
host ssmpr postgres 127.0.0.1/32 trust
host all all ::1/128 md5
```

Сохранить и применить новую конфигурацию сервера баз данных:

```
$ sudo service postgresql restart
```

Создать базу данных, пользователя и установить расширения для сервера баз данных, а также импортировать дампы из файла `ssmp.sql`:

Определить имена: базы – `ssmp`, имя пользователя – `ssmp`, пароль – `ssmp`.

Авторизоваться в сервере баз данных:

```
$ psql -U postgres
```

При входе в консоль управления сервером баз данных изменится вид приглашения к вводу (в консоли):

```
postgres=#
```

Установить необходимые расширения для PostgreSQL:

```
postgres=#CREATE EXTENSION postgis;
postgres=#CREATE EXTENSION fuzzystrmatch;
postgres=#CREATE EXTENSION postgis_tiger_geocoder;
```

Создать базу данных с именем `ssmp`:

```
postgres=#CREATE DATABASE ssmp;
```

Создать пользователя и назначить его владельцем созданной БД с именем `ssmp`:

```
postgres=#CREATE USER ssmp WITH PASSWORD 'strongpassword';
```

Добавить пользователя `ssmp` с ролью `superuser`:

```
postgres=#ALTER ROLE ssmp WITH superuser;
```

Добавить привилегии на базу данных `ssmp` для пользователя `ssmp`:

```
postgres=#GRANT ALL PRIVILEGES ON DATABASE ssmp TO ssmp;
```

Обновить список привилегий в глобальном схеме:

```
postgres=#FLUSH PRIVILEGES
```

Выполнить импорт дампа базы данных в PostgreSQL, для этого выполнить команду:

```
$ psql -U ssmpp ssmpp < ssmpp.sql
```

18) Скопировать конфигурационный файл менеджера задач в директорию /etc/supervisor/conf.d/ и переименовать в ssmpp.conf:

```
[program:ssmpp-riams-send-KV]
```

```
command=bash -c 'while true; do php /data/www/ssmpp/artisan riams:calls:send calls;sleep 60; done'
```

```
user=www-data
```

```
[program:ssmpp-riams-send-KT]
```

```
command=bash -c 'while true; do php /data/www/ssmpp/artisan riams:calls:send ;sleep 60; done'
```

```
user=www-data
```

```
[program:ssmpp-riams-status]
```

```
command=bash -c 'while true; do php /data/www/ssmpp/artisan riams:calls:status ;sleep 60; done'
```

```
user=www-data
```

```
[program:default-queue]
```

```
command=php7.3 /data/www/ssmpp/artisan queue:listen --tries=1 --timeout=1800 --memory=2048
```

```
user=www-data
```

```
process_name=%(program_name)s_%(process_num)s
```

```
numprocs=3
```

```
autostart=true
```

```
autorestart=true
```

```
[program:mileage-queue]
```

```
command=php7.3 /data/www/ssmpp/artisan queue:listen --tries=1 --timeout=1800 --memory=2048 --queue=mileage
```

```
user=www-data
```

```
process_name=%(program_name)s_%(process_num)s
```

```
numprocs=1
```

```
autostart=true
```

```
autorestart=true
```

```
[program:ssmpp-pinger]
```

```
command=bash -c 'while true; do php7.3 /data/www/ssmp/artisan ssmp:pinger;sleep 5;done'  
user=www-data
```

```
[program:ssmp-cron]
```

```
command=bash -c 'while true; do php7.3 /data/www/ssmp/artisan cron:run ;sleep 30;done'  
user=www-data
```

```
[program:ssmp-schedule]
```

```
command=bash -c 'while true; do php7.3 /data/www/ssmp/artisan schedule:run ;sleep 55;done'  
user=www-data
```

```
[program:ssmp-ktstatus]
```

```
command=bash -c 'while true; do php7.3 /data/www/ssmp/artisan ssmp:check-kt-status-  
confirm ;sleep 5;done'  
user=www-data
```

```
[program:ssmp-pusher]
```

```
command=php7.3 /data/www/ssmp/artisan ssmp:run-pusher  
user=www-data  
autorestart=true
```

```
[program:delete-notify]
```

```
command=bash -c 'while true; do php7.3 /data/www/ssmp/artisan ssmp:delete-notify ;sleep  
3600;done'  
user=www-data
```

```
[program:soft-delete-draft-kt]
```

```
command=bash -c 'while true; do php7.3 /data/www/ssmp/artisan ssmp:soft-delete-draft-kt ;  
sleep 3700;done'  
user=www-data
```

```
[program:ssmp-reports-cache]
```

```
command=php7.3 /data/www/ssmp/artisan ssmp:run-reports-cache  
user=www-data
```

```
[program:ssmp-telematics]
```

```
command=php7.3 /data/www/ssmp/artisan ssmp:run-telematics  
autostart=true  
autorestart=true
```

user=www-data

19) Перейти на уровень выше корневой директории веб-сервера /data/www/http/ и выполнить:

```
$ php composer.phar install
```

Выполнить миграции с помощью команды:

```
$ php artisan migrate
```

ИС СМП установлена и полностью готова к работе в штатном режиме.

20) Производим установку сервисов СШ ЕЦП СМП, для этого производим разархивацию zip-файла с модулями:

```
$ sudo unzip esp_smp.zip ./Modules
```

В результате распаковки получаем набор сервисов в директории /data/www/ssmp/Modules

21) Производим очистку кеша сервера приложений:

```
$ php artisan cache-clear
```

22) Копируем в конфигурационный файл менеджера задач следующие программы /etc/supervisor/conf.d/ssmp.conf:

```
[program:112-queue]
```

```
command=php7.3 /data/www/http/artisan queue:listen --tries=1 --timeout=1800 --memory=2048 --queue=sendTo112
```

```
[program:tfoms-queue]
```

```
command=php7.3 /data/www/http/artisan queue:listen --tries=1 --timeout=1800 --memory=512 --queue=tfoms
```

```
user=www-data
```

```
process_name=%(program_name)s_%(process_num)s
```

```
numprocs=3
```

```
autostart=true
```

```
autorestart=true
```

```
[program:ssmp-rmis-send-KV]
```

```
command=bash -c 'while true; do php7.3 /data/www/http/artisan riams:calls:send calls;sleep 60;done'
```

```
user=www-data
```

```
[program:ssmp-rmis-send-KT]
```

```
command=bash -c 'while true; do php7.3 /data/www/http/artisan
riams:calls:send;sleep 60;done'
user=www-data
```

```
[program:ssmp-rmis-status]
command=bash -c 'while true; do php7.3 /data/www/http/artisan
riams:calls:status;sleep 60;done'
user=www-data
```

```
[program:ssmp-sync]
command=bash -c 'while true; do php7.3 /data/www/http/artisan ssmp:replication-
server-sync;sleep 60;done'
user=www-data
```

```
[program:ssmp-server]
command=bash -c 'while true; do php7.3 /data/www/http/artisan
ssmp:replication-server;sleep 60;done'
user=www-data
```

```
[program:ssmp-client]
command=bash -c 'while true; do php7.3 /data/www/http/artisan
ssmp:replication-client;sleep 60;done'
user=www-data
```

```
[program:ssmp-sip-stats]
command=bash -c 'while true; do php7.3 /data/www/http/artisan ssmp:cron:sip-
stats-sync;sleep 60;done'
user=www-data
```

23) Производим настройку сервера приложений:

23.1) Открываем файл с переменными сервера приложений /data/www/ssmp/.env

23.2) добавляем следующие строки:

```
MODULES_CACHE_ENABLED=true - активация кеша для сервисов ЕЦП
СМП;
SERVER_112_STATUS=true – активация сервера приема происшествий от
системы 112;
CLIENT_112_STATUS=true – активация клиента обмена с системой 112;
CLIENT_112_PROTOCOL=http – указание протокола сетевого обмена;
CLIENT_112_SERVER= адрес сервера 112;
CLIENT_112_SERVER_PORT= порт соединения с сервером 112;
CLIENT_112_TIMEOUT= таймаут соединения в секундах;
CLIENT_112_PATH=/location сервиса обмена системы 112;
FB_M_RMIS=true – активация модуля интеграции с РМИС;
FB_M_RMIS_PROTOCOL=http;
```

FB\_M\_RMIS\_HOST=ip адрес системы РМИС;  
FB\_M\_RMIS\_PATH=/location сервиса обмена данными (API);  
REPLICATION\_DISABLED = false – активирует обмен между подсистемами ИС СМП;  
REPLICATION\_MASTER = true – указывается на сервере сбора данных;  
REPLICATION\_CLIENT = true – указывается на сервере отправки данных;  
FB\_M\_SIP = true – активация модуля взаимодействия с АТС;  
SIP\_STATS = true – получение статистике по группам операторов КЦ;  
FB\_M\_TFOMS285=true – активация модуля подготовки счетов-реестров ТФОМС;  
TFOMS\_TF\_CODE= код региона (01,02,03 и т.д.);  
FB\_M\_EXTERNAL\_REQUESTS=true – активация внешних вызовов для обмена с карточными платформами.

24) Производим миграцию полей базы данных для ЕЦП СМП:  
\$ php artisan module:migrate.

СШ ЕЦП СМП установлена и полностью готова к работе в штатном режиме.

#### 4 ПРОВЕРКА ПРОГРАММЫ

Для проверки программы необходимо выполнить следующие действия:

Открыть в браузере новую страницу и в поле адрес ввести:  
<http://192.168.1.100>. Открывается форма авторизации Программы.

## 5 СООБЩЕНИЯ АДМИНИСТРАТОРУ

Сообщения администратору подразделяются на:

- Диагностические сообщения;
- Сообщения об ошибках.

Все сообщения протоколируются в журналах логов, расположенных в папке /data/www/logs/.

### 5.1 Сообщения об ошибках

Сообщения об ошибках сервера, выдаваемые системой программисту, приведены в Таблица 4.

Таблица 4 – Сообщения об ошибках сервера

№ п/п	Ошибка	Условия выдачи сообщения
1	Ошибки взаимодействия с внешними ИС	
1.1	ОШИБКА: Unable to complete network request to host «127.0.0.1».Failed to establish a connection. Подключение не установлено, т.к. конечный компьютер отверг запрос на подключение	Сообщение появляется при невозможности соединения с ИС СМП
1.2	ОШИБКА: Unable to complete network request to host «127.0.0.1».Failed to establish a connection. Подключение не установлено, т.к. конечный компьютер отверг запрос на подключение	Сообщение появляется при невозможности соединения с Системой-112
1.3	ОШИБКА: Unable to complete network request to host «127.0.0.1».Failed to establish a connection. Подключение не установлено, т.к. конечный компьютер отверг запрос на подключение	Сообщение появляется при невозможности соединения с РМИС
1.4	ОШИБКА: Unable to complete network request to host «127.0.0.1».Failed to establish a connection. Подключение не установлено, т.к. конечный компьютер отверг запрос на подключение	Сообщение появляется при невозможности соединения с ЕИР ТФОМС
1.5	ОШИБКА: Unable to complete network request to host «127.0.0.1».Failed to establish a connection. Подключение не установлено, т.к. конечный компьютер отверг запрос на подключение	Сообщение появляется при невозможности соединения с ПО сервисной платформы цифровой АТС
2	Ошибки конфигурационного файла	
2.1	ОШИБКА: Ошибка сокета 10049 при выполнении bind. Требуемый адрес для своего контекста неверен	Неверный IP адрес или порт в конфигурационном файле



## **5.2 Протоколирование ошибок и предупреждений**

Все ошибки записываются в лог с началом записи «ОШИБКА».

Все предупреждения записываются в лог с началом записи «ПРЕДУПРЕЖДЕНИЕ».