

**САП «CUBALINK»**

Руководство по развертыванию

Симферополь 2024

**Оглавление**

[1. Подготовка инфраструктуры 3](#_gjdgxs)

[2. Состав дистрибутива 4](#_30j0zll)

[3. Установка ПО 5](#_1fob9te)

[3.1. Настройка СУБД 5](#_3znysh7)

[3.2. Настройка веб-сервера 5](#_2et92p0)

[3.3. Создание пользователей 6](#_tyjcwt)

# Подготовка инфраструктуры

Для развертывания ПО необходим сервер (1 ед.) с различными системными приложениями и службы, такие как интерпретатор языка PHP с набором расширений, WEB-сервер, система управления базами данных и прочие.

В качестве операционной системы на сервере должна использоваться Debian 12.

Необходимо обратить внимание, что операционная система должна быть сконфигурирована с корректным часовым поясом и локалью. От этого зависит правильность отображения информации и корректность сортировки.

Список требуемых расширений PHP содержит все расширения, не входящие в состав ядра PHP. Часть из этих расширений может поставляться вместе с основным пакетом PHP либо в составе пакета php-common, другая же часть должна быть установлена дополнительно.

* PHP: 8.3
* Расширения PHP: ctype, gd, json, libxml, mbstring, openssl, pdo, pdo\_pgsql, posix, simplexml, snmp, sockets, zlib, pcntl
* Дополнительные расширения PHP: ldap, soap
* Python: 3.9+ (желательно 3.11)
* Модули Python: pip, venv
* PostgreSQL: 12+ (желательно 16)
* Redis: 5+ (желательно 7)
* RabbitMQ: 3.10+ (желательно 3.13)
* Supervisor

Системные требования к серверу: RAM 16ГБ+, свободное место на жестком диске 30ГБ+.

# Основные договорённости

Каталог, в который устанавливается ПО «*/var/www/cubalink*». Внутри этого каталога должен быть подкаталог public, который является корневым каталогом для WEB-сервера. Если планируется использовать другой каталог, то необходимо не забывать вносить соответствующие исправления во все примеры из данного руководства.

Необходимо создать каталог:

**sudo mkdir -p /var/www/cubalink/public**

**sudo chown -R www-data:www-data /var/www/cubalink**

При необходимости нужно выделить доменное имя для ПО. В примерах ниже используется cubalink.mycompany.ru, но его необходимо будет заменить на свое. Требуется прописать это доменное имя в /etc/hosts, если оно отличается от имени хоста, на котором производится установка.

# Установка необходимых компонентов

Сперва необходимо установить утилиты, которые будут необходимы далее в процессе установки. Можно копировать строки по одной и вставлять их в командную строку операционной системы.

Следующие строки необходимо скопировать и вставить целиком:

sudo tee /etc/apt/sources.list.d/contrib-non-free.list << EOL

deb http://deb.debian.org/debian/ $(lsb\_release -sc) contrib non-free non-free-firmware

EOL

sudo apt update

sudo apt full-upgrade -y

sudo apt install -y gnupg ca-certificates lsb-release debian-archive-keyring debian-keyring libsnmp-dev snmp-mibs-downloader

### PostgreSQL

Рекомендуется использовать последнюю версию PostgreSQL, так как это всегда положительно сказывается на скорости работы. Необходимо добавить в систему репозиторий Postgres:

sudo install -d /usr/share/postgresql-common/pgdg

sudo curl -o /usr/share/postgresql-common/pgdg/apt.postgresql.org.asc --fail https://www.postgresql.org/media/keys/ACCC4CF8.asc

sudo sh -c 'echo "deb [signed-by=/usr/share/postgresql-common/pgdg/apt.postgresql.org.asc] https://apt.postgresql.org/pub/repos/apt $(lsb\_release -cs)-pgdg main" > /etc/apt/sources.list.d/pgdg.list'

sudo apt update

Перед установкой PostgreSQL в операционной системе должна быть установлена местная локаль и корректный часовой пояс. Это можно сделать и позже, но будет намного проще, если это будет сделано до установки PostgreSQL. Необходимо выполнить следующую команду, чтобы убедиться, что нужная локаль присутствует:

locale -a

Если среди перечисленных нет местной локали, то необходимо установить ее. Для этого достаточно отредактировать файл /etc/locale.gen убрав комментарии перед строкой (строками) с нужной локалью, а затем выполнить команду:

sudo locale-gen

Если нужно изменить локаль по умолчанию, то необходимо выполнить:

sudo dpkg-reconfigure locales

Чтобы изменить часовой пояс, необходимо выполнить:

sudo timedatectl set-timezone Europe/Moscow

Теперь можно перейти к установке PostgreSQL:

sudo apt install -y postgresql-16 postgresql-16-postgis-3

Необходимо создать пользователя и базу данных для ПО. В примере ниже создается пользователь с именем cubalink и база данных с таким же именем. Если локаль не ru\_RU, необходимо изменить команду. Затем к базе данных подключается расширение PostGis, необходимое для работы с географическими данными. После выполнения первой строки, необходимо будет дважды ввести пароль для нового пользователя – необходимо записать этот пароль, он понадобится далее для установки ПО

sudo -u postgres createuser cubalink -P

sudo -u postgres createdb -e -E "UTF-8" -l "ru\_RU.UTF-8" -O cubalink -T template0 cubalink

sudo -u postgres psql -d cubalink -c "CREATE EXTENSION postgis"

Redis

Стандартный репозиторий Debian 12 включает версию Redis 7.0.15. Эта версия подходит для работы САП CUBALINK.

sudo apt install -y redis-server

По умолчанию Redis принимает подключения без пароля, но настоятельно рекомендуется установить пароль. Так как пароль передается в открытом виде (Redis не предусматривает шифрования, т.к. делает упор на скорость обработки запросов и лишние этапы вроде шифрования не применяются), нужно создать действительно длинный и сложный пароль. Далее требуется вместо слова MYPASSWORDHERE подставить пароль:

sudo sed -i 's@^.\*requirepass .\*@requirepass MYPASSWORDHERE@g' /etc/redis/redis.conf

Также необходимо отключить таймаут (на некоторых версиях Redis он включен):

sudo sed -i 's@^timeout .\*@timeout 0@' /etc/redis/redis.conf

Требуется перезапустить Redis и убедитесь, что он работает (в ответ должен вернуться PONG):

sudo systemctl restart redis

redis-cli -h 127.0.0.1 -p 6379 -a MYPASSWORDHERE ping

RabbitMQ

Стандартный репозиторий Debian 12 включает старую версию RabbitMQ 3.10. Можно использовать ее — она поддерживается. Однако, эта версия относительно старая и лучшим вариантом будет установить новую версию из альтернативных репозиториев. На [официальном сайте RabbitMQ](https://www.rabbitmq.com/install-debian.html) находится достаточно подробная информация по установке RabbitMQ для каждой операционной системы.

Следующий блок требуется скопировать и вставить целиком в командную строку

curl -1sLf https://dl.cloudsmith.io/public/rabbitmq/rabbitmq-erlang/gpg.E495BB49CC4BBE5B.key | sudo gpg --dearmor | sudo tee /usr/share/keyrings/rabbitmq-erlang-archive-keyring.gpg > /dev/null

curl -1sLf https://dl.cloudsmith.io/public/rabbitmq/rabbitmq-server/gpg.9F4587F226208342.key | sudo gpg --dearmor | sudo tee /usr/share/keyrings/rabbitmq-server-archive-keyring.gpg > /dev/null

sudo tee /etc/apt/sources.list.d/rabbitmq.list <<EOF

deb [signed-by=/usr/share/keyrings/rabbitmq-erlang-archive-keyring.gpg] https://dl.cloudsmith.io/public/rabbitmq/rabbitmq-erlang/deb/debian $(lsb\_release -cs) main

deb-src [signed-by=/usr/share/keyrings/rabbitmq-erlang-archive-keyring.gpg] https://dl.cloudsmith.io/public/rabbitmq/rabbitmq-erlang/deb/debian $(lsb\_release -cs) main

deb [signed-by=/usr/share/keyrings/rabbitmq-server-archive-keyring.gpg] https://dl.cloudsmith.io/public/rabbitmq/rabbitmq-server/deb/debian $(lsb\_release -cs) main

deb-src [signed-by=/usr/share/keyrings/rabbitmq-server-archive-keyring.gpg] https://dl.cloudsmith.io/public/rabbitmq/rabbitmq-server/deb/debian $(lsb\_release -cs) main

EOF

sudo apt update

Необходимо убедится, что в /etc/hosts есть запись для имени хоста. Отсутствие такой записи (например, если было изменено имя хоста после инсталляции) является основной ошибкой, возникающей при установке RabbitMQ. Так что лучше дополнительно убедится, что запись соответствующая имени хосту имеется.

sudo apt install -y erlang-base \

 erlang-asn1 erlang-crypto erlang-eldap erlang-ftp erlang-inets \

 erlang-mnesia erlang-os-mon erlang-parsetools erlang-public-key \

 erlang-runtime-tools erlang-snmp erlang-ssl \

 erlang-syntax-tools erlang-tftp erlang-tools erlang-xmerl

sudo apt install rabbitmq-server -y --fix-missing

Следующие две строки необходимо выполнять по одной за раз.

sudo rabbitmq-plugins enable rabbitmq\_management --offline

sudo rabbitmq-plugins enable rabbitmq\_web\_stomp --offline

Необходимо скопировать и вставить целиком следующий блок:

sudo tee /etc/rabbitmq/rabbitmq.conf << EOF

listeners.tcp.default = 5672

web\_stomp.port = 15674

web\_stomp.cowboy\_opts.max\_keepalive = 60

EOF

Теперь требуется создать пользователей. Рекомендуется использовать три различных пользователя: для администрирования сервера (полные права), для САП CUBALINK и модулей (полные права но без административных) и пользователя web-stomp для websocket (минимальные права на чтение определенных объектов брокера). Можног создать только два: администратора, который также будете использоваться для САП CUBALINK, и пользователя для websocket. Это обязательно должны быть минимум два разных пользователя, так как пароль для websocket пользователя передается в браузер и может быть легко прочитан пользователем. Далее будет показан пример для рекомендуемых трех пользователей.

Необходимо создать пользователя для администрирования RabbitMQ. В данном примере используется имя пользователя admin и пароль пароль\_администратора. Пользователю сразу после создания присваивается тэг administrator, наделяющий пользователя максимальными правами администратора и затем устанавливаются разрешения для vhost / позволяющие полный доступ к конфигурированию, записи и чтению всего в пределах этого vhost\*.

sudo rabbitmqctl add\_user "admin" "пароль\_администратора"

sudo rabbitmqctl set\_user\_tags "admin" "administrator"

sudo rabbitmqctl set\_permissions -p "/" "admin" ".\*" ".\*" ".\*"

Необходимо создать пользователя, от имени которого будут работать САП CUBALINK и модули. Пример для имени пользователя cubalink (наделять правами администратора этого пользователя не требуется):

sudo rabbitmqctl add\_user "cubalink" "пароль\_системы"

sudo rabbitmqctl set\_user\_tags "cubalink" "monitoring"

sudo rabbitmqctl set\_permissions -p "/" "cubalink" ".\*" ".\*" ".\*"

Необходимо создать пользователя WebSTOMP. Он понадобится для использования уведомлений через WebSocket. Вместо websock-user можно указать другое имя пользователя. Вместо пароль\_websocket требуется указать свой пароль, но этот пароль будет передаваться в открытом виде в браузер, так что не допускается делать его похожим на остальные пароли:

sudo rabbitmqctl add\_user "websock-user" "пароль\_websocket"

sudo rabbitmqctl set\_permissions -p "/" "websock-user" "^erp-stomp:id-.\*" "" "^erp-stomp:id-.\*"

Внимание! Имя пользователя и пароль WebSTOMP нужно будет указать в настройках в интерфейсе САП CUBALINK (Меню: Настройки - Основные - Websocket).

\* vhost - это виртуальный хост внутри RabbitMQ, позволяющий разграничить различные варианты использования одного и того же сервера разными приложениями. Как, например, разные базы данных на одном сервере СУБД. vhost по умолчанию имеет имя / и почти всегда этого достаточно. Но если планируется, например, запустить несколько копий САП CUBALINK на одном сервере, то для каждой копии нужно будет создать свой vhost и, соответственно, пользователей для него.

Необходимо перезапустить службу rabbitmq:

sudo systemctl restart rabbitmq-server

Одна из команд, приведенных выше, устанавливает в RabbitMQ модуль управления, предоставляющий удобный WEB-интерфейс для мониторинга, диагностики и управления сервером RabbitMQ. При помощи этого модуля можно осуществлять мониторинг сервера, следить за количеством сообщений в очередях и другими состояниями. Всё это можно делать используя консольную утилиту rabbitmqctl, но использование WEB-интерфейса может быть значительно наглядней и удобней.

По умолчанию WEB-интерфейс управления доступен по адресу [http://cubalink.mycompany.ru:15672](http://userside.mycompany.com:15672/). Рекомендуется использовать фаервол, чтобы ограничить доступ к этому интерфейсу управления.

Подробнее о модуле управления можно [прочитать на официальном сайте RabbitMQ](https://www.rabbitmq.com/management.html).

**PHP**

Альтернативный репозиторий

sudo curl -sSLo /usr/share/keyrings/deb.sury.org-php.gpg https://packages.sury.org/php/apt.gpg

sudo sh -c 'echo "deb [signed-by=/usr/share/keyrings/deb.sury.org-php.gpg] https://packages.sury.org/php/ $(lsb\_release -sc) main" > /etc/apt/sources.list.d/php.list'

sudo apt update

Необходимо выполнить следующие команды:

sudo apt install -y php8.3-{fpm,cli,common,curl,intl,mbstring,opcache,pgsql,readline,xml,zip,snmp,gd}

Далее приведены команды, вносящие изменения в файлы конфигурации.

sudo sed -i "s@^;date.timezone.\*@date.timezone = $(cat /etc/timezone)@" /etc/php/8.3/fpm/php.ini

sudo sed -i "s@^;date.timezone.\*@date.timezone = $(cat /etc/timezone)@" /etc/php/8.3/cli/php.ini

sudo sed -i "s@;cgi.fix\_pathinfo=1@cgi.fix\_pathinfo=0@" /etc/php/8.3/fpm/php.ini

sudo sed -i "s@post\_max\_size = 8M@post\_max\_size = 100M@" /etc/php/8.3/fpm/php.ini

sudo sed -i "s@upload\_max\_filesize = 2M@upload\_max\_filesize = 100M@" /etc/php/8.3/fpm/php.ini

sudo sed -i "s@max\_execution\_time.\*@max\_execution\_time = 300@" /etc/php/8.3/fpm/php.ini

sudo sed -i "s@^;request\_terminate\_timeout =.\*@request\_terminate\_timeout = 300@" /etc/php/8.3/fpm/pool.d/www.conf

sudo systemctl restart php8.3-fpm

**NGINX**

Версия NGINX, поставляемая в репозитории Debian, полностью подходит для использования и нет необходимости в добавлении альтернативного репозитория.

sudo apt install -y nginx

sudo sed -i "s@^user.\*;@user www-data www-data;@" "/etc/nginx/nginx.conf"

sudo systemctl restart nginx

Следующий пример настройки подразумевает, что САП CUBALINK будет установлена в стандартный каталог системы /var/www/cubalink. Если требуется установить САП CUBALINK в другой каталог, необходимо исправлять его во всех последующих примерах.

Далее необходимо отредактировать файл /etc/nginx/conf.d/default.conf. Либо удалить его и создать файл с именем /etc/nginx/conf.d/cubalink.conf. В любом случае, не зависимо от выбора, содержимое файла должно быть следующим (вместо cubalink.mycompany.ru необходимо указать свое доменное имя):

server {

 listen 80 default\_server;

 server\_name cubalink.mycompany.ru;

 charset utf-8;

 client\_max\_body\_size 100M;

 access\_log /var/log/nginx/cubalink-access.log;

 error\_log /var/log/nginx/cubalink-error.log;

 root /var/www/cubalink/public;

 index index.php;

 location = /favicon.ico { access\_log off; log\_not\_found off; }

 location = /robots.txt { access\_log off; log\_not\_found off; }

 location / {

 try\_files $uri $uri/ /index.php$is\_args$args;

 }

 location ~\* ^.+\.(css|js|ogg|ogv|svg|svgz|eot|otf|woff|mp4|ttf|rss|atom|jpg|jpeg|gif|png|ico|zip|tgz|gz|rar|bz2|doc|xls|exe|ppt|tar|mid|midi|wav|bmp|rtf)$ {

 access\_log off;

 log\_not\_found off;

 expires max;

 add\_header Pragma public;

 add\_header Cache-Control "public";

 }

 location ~ \.php$ {

 try\_files $uri =404;

 fastcgi\_split\_path\_info ^(.+\.php)(/.+)$;

 fastcgi\_pass unix:/run/php/php8.3-fpm.sock;

 fastcgi\_index index.php;

 fastcgi\_param SCRIPT\_FILENAME $document\_root$fastcgi\_script\_name;

 fastcgi\_read\_timeout 300;

 include fastcgi\_params;

 }

 location /ws {

 proxy\_pass http://127.0.0.1:15674/ws;

 proxy\_http\_version 1.1;

 proxy\_set\_header Upgrade $http\_upgrade;

 proxy\_set\_header Connection "Upgrade";

 proxy\_set\_header Host $host;

 }

 location ~ /\.ht { deny all; }

}

Необходимо проверить конфигурацию и если она в порядке перезагрузить nginx:

sudo nginx -t && sudo nginx -s reload

 **Python**

Debian 12 в стандартном репозитории содержит Python 3.11 - этого вполне достаточно, поэтому никакие альтернативные репозитории не нужны.

Установка необходимых пакетов:

sudo apt install -y python3-dev python3-pip python3-venv libffi-dev pkg-config

 **Supervisor**

sudo apt install -y supervisor

# Загрузка и запуск инсталлятора

Требуется перейти в каталог ПО

cd /var/www/cubalink

Загрузить скрипт-инсталлятор

sudo -u www-data php -r "copy('[https://d.cubalink.ru/install'](https://d.cubalink.ru/install%27), 'install.phar');"

Запустить инсталлятор

sudo -u www-data php install.phar install

В процессе работы инсталлятор проверяет соответствие техническим требованиям, запрашивает параметры доступа и проверяет соединения со службами сервера.

По окончанию работы инсталлятора будет выведено сообщение об успешной установке.

Если установка была проведена не от имени пользователя веб-сервера, то обязательно после окончания установки требуется сделать его владельцем всех файлов САП CUBALINK рекурсивно!

sudo chown -R www-data:www-data /var/www/cubalink

sudo chmod -R u=rwX,g=rwX,o=r /var/www/cubalink

# Настройка после установки

Конфигурация системных служб:

После установки необходимо скопировать примеры конфигурационных файлов системных служб и, при необходимости, настроить их (например, указать путь к каталогу cubalink, если он отличается от /var/www/cubalink). Требуется выполнить команды:

sudo cp etc/us-core-worker.conf-example /etc/supervisor/conf.d/us-core-worker.conf

sudo cp microservice/poller/etc/usm\_poller.conf-example /etc/supervisor/conf.d/usm\_poller.conf

sudo cp etc/logrotate-example /etc/logrotate.d/cubalink

sudo cp microservice/poller/etc/logrotate-example /etc/logrotate.d/usm\_poller

sudo cp etc/crontab-example /etc/cron.d/cubalink

Установка необходимых зависимостей для usm\_poller:

Необходимо создать виртуальное окружение venv для python и установить зависимые модули, как показано далее:

cd /var/www/cubalink/microservice/poller

sudo -H python3 -m venv venv

sudo -H venv/bin/pip install -U pip

sudo -H venv/bin/pip install -U -r requirements.txt

Запуск супервизора:

Супервизор контролирует работу служб, указанных в его конфигурации. Для запуска супервизора необходимо выполнить:

sudo systemctl restart supervisor

После запуска супервизора, спустя несколько секунд можно понаблюдать за состоянием всех контролируемых им служб. Все службы должны быть в состоянии RUNNING:

sudo supervisorctl status

# Сразу после установки

На этом установка САП CUBALINK завершена. Теперь необходимо выполнить следующие шаги:

Открыть в браузере страницу системы http://cubalink.mydomain.ru/ и выполнить вход (по умолчанию имя пользователя: Admin, пароль: 1234).

Выполнить настройку Websocket в разделе: Настройка - Основная - WebSocket. Необходимо включить и вписать имя пользователя и пароль WebStomp пользователя, которой ранее был создан в разделе RabbitMQ.