



[dev: add test sso installation, fix: realm import anf default admin creation, fix: documentation](#)

Popov Dmitry создал 1 месяц назад

vats.md 16,51 КиБ

Требования к системе

Для запуска системы требуется выделенная физическая или виртуальная машина, с запущенной на ней операционной системы Linux (например Debian 12) с установленными пакетами обеспечивающими запуск docker контейнеров. Версия docker должна быть не ниже 26. Так же для упрощения запуска рекомендуется установить расширение docker compose.

С точки зрения физических ресурсов необходимо:

- Процессор должен поддерживать набор инструкций x86-64
- Минимум 2 процессорных ядра производительностью не ниже Intel Xeon [E5-2699v3@2.30GHz](#)
- 4Gi оперативной памяти
- не менее 30Gi выделенного места жёсткого диска.

Базы данных

Для работы системы необходима работающие серверы баз данных на базе PostgreSQL Server версии 16.

```
Мы рекомендуем использовать 2 сервера баз данных: один для хранения конфигураций и управления системой, второй для хранения
```

На сервере БД, выделенном для хранения конфигураций и управления системой необходимо создать 3 БД и 2 учётные записи для доступа к ним:

наименование	назначение
vats_pbx	конфигурация телефонной станции
vats_mgmt	управление ВАС
vats_keycloak	управление пользователями ВАС

На сервере для хранения статистики создаём БД и учётную запись для доступа к ней:

наименование	назначение
vats_cdr	хранение статистической информации

```
У баз данных **vats_pbx** и **vats_cdr** должны быть отключены параметры *standard_conforming_strings* и *escape_string
```

Установка системы

Система представляет собой набор контейнеров, для которых нужно подготовить конфигурацию и осуществить запуск.

Предполагаем, что конфигурацию и параметризацию запуска будем производить в каталоге `/srv/vats`. В дальнейшем, если не указано обратного, подразумевается что действия производит именно в этом каталоге.

Так как система поставляется в виде набора контейнеров, то для упрощения запуска используется единый `docker-compose.yml` файл, который поставляется вместе с инструкцией. Мы подразумеваем, что инженер осуществляющий запуск имеет квалификацию, позволяющую ему вносить изменения в `docker-compose.yml` файл.

Так же в примерах конфигурирования подключений к БД указываются пользователи и имена баз данных из *примера создания баз данных*

Подготовка к установке

Перед установкой системы нужно произвести сбор данных и заполнить таблицу.

```
В качестве примера, в таблице приведены значения для сервера, запущенного в cloud.ru
```

Параметр	Значение	Описание
SERVER_IP	172.16.1.2	IP адрес сервера, который ему непосредственно назначен, можно посмотреть с помощью команды <code>ip addr show</code>
EXTERNAL_IP	37.18.100.149	IP адрес, по которому к серверу будут подключаться клиенты. Может совпадать с SERVER_IP.
DEDICATED_NET	172.31.0.0/24	Выделенная локальная подсеть, нужная для внутрисервисного обмена. Крайне желательно, чтобы она не пересекалась с другими подсетями, которые есть в существующей сетевой инфраструктуре
RTP_PORT_START	11000	Начальный номер диапазона udp портов, предназначенных для передачи голосового трафика. В случае, если EXTERNAL_IP отличается от SERVER_IP, то данный диапазон должен быть проброшен.
RTP_PORT_END	11999	Конечный номер диапазона udp портов, предназначенных для передачи голосового трафика. В случае, если EXTERNAL_IP отличается от SERVER_IP, то данный диапазон должен быть проброшен.
PBX_CONFIG_DATABASE_HOST	psql-server.local	Адрес PostgreSQL сервера, на котором будет храниться конфигурация BATC
PBX_CONFIG_DATABASE_NAME	vats_pbx	Имя БД хранения конфигурации BATC
PBX_CONFIG_DATABASE_USER	vats_pbx	Имя пользователя для доступа к БД хранения конфигурации BATC
PBX_CONFIG_DATABASE_PASSWORD	xxxxxxx	Пароль пользователя PBX_CONFIG_DATABASE_USER
CDR_DATABASE_HOST	psql-server.local	Адрес PostgreSQL сервера, где будет храниться статистика BATC
CDR_DATABASE_NAME	vats_cdr	Имя БД хранения статистики BATC
CDR_DATABASE_USER	vats_cdr	Имя пользователя для доступа к БД хранения статистики BATC
CDR_DATABASE_PASSWORD	xxxxxxx	Пароль пользователя CDR_DATABASE_USER
KEYC_DATABASE_HOST	psql-server.local	Адрес PostgreSQL сервера, где будут храниться данные сервиса управления пользователями
KEYC_DATABASE_NAME	vats_keycloak	Имя базы данных для использования сервисом управления пользователями
KEYC_DATABASE_USER	vats_keycloak	Имя пользователя для доступа к базе данных KEYC_DATABASE_NAME
KEYC_DATABASE_PASSWORD	xxxxxxx	Пароль пользователя KEYC_DATABASE_USER
CONTROL_PLANE_DB_HOST	psql-server.local	Адрес PostgreSQL сервера, где будут храниться данные сервиса управления BATC
CONTROL_PLANE_DB_NAME	vats_mgmt	Имя базы данных для использования сервисом управления BATC
CONTROL_PLANE_DB_USER	vats_mgmt	Имя пользователя для доступа к базе данных CONTROL_PLANE_DB_NAME
CONTROL_PLANE_DB_PASSWORD	xxxxxxx	Пароль пользователя CONTROL_PLANE_DB_USER

В дальнейших шагах нужно будет значения из данной таблицы заменять на соответствующие значения из данной таблицы. Например если в тексте будет написано

```
RTP_PORT_START=%RTP_PORT_START%
```

то данную строчку следует отредактировать, заменив `%RTP_PORT_START%` на значение из таблицы:

```
RTP_PORT_START=11000
```

Аутентификация в Docker registry

для аутентификации в docker registry нужно выполнить команду

```
docker login docker.1c-service.com
```

и ввести выданные учётные данные

Телефонная часть системы

Задание конфигурации доступа к БД

Создаём файл `pbx_db.env`, в котором прописываем параметры доступа к базам данных хранения конфигурации *BATC* и хранения статистики *BATC* (не забываем заменить значения на соответствующие из таблицы):

```
# Параметры подключения к хранению конфигурации BATC
PBX_DB_NAME=%PBX_CONFIG_DATABASE_NAME%
PBX_DB_SERVER=%PBX_CONFIG_DATABASE_HOST%
PBX_DB_USER=%PBX_CONFIG_DATABASE_USER%
PBX_DB_PASSWORD=%PBX_CONFIG_DATABASE_PASSWORD%

# Параметры подключения к хранению статистика BATC
CDR_DB_NAME=%CDR_DATABASE_NAME%
CDR_DB_SERVER=%CDR_DATABASE_HOST%
CDR_DB_USER=%CDR_DATABASE_USER%
CDR_DB_PASSWORD=%CDR_DATABASE_PASSWORD%

# Параметры внутреннего взаимодействия
CRON_REMOTE_USER=cron
CRON_REMOTE_PASSWORD=secret!!!123
```

Установка схем данных и загрузка необходимых данных

Запускаем установку схем баз данных:

```
docker run --rm -ti -v ./pbx_db.env:/.env:ro docker.1c-service.com/vats/rhombus/standalone/install:1.0.0-dev3
```

Внимание! Запуск установки схем нужно производить только 1 раз!

Запуск системы

Создание конфигурационных файлов

в каталоге `dist` лежат примеры конфигурационных файлов, которые можно использовать в качестве шаблона

Создаём файл `.env` в котором опишем параметры запуска контейнеров с приложениями. не забываем заменять значения параметров на соответствующие из таблицы.

```
RTP_PORT_START=%RTP_PORT_START%
RTP_PORT_END=%RTP_PORT_END%
EXTERNAL_IP=%EXTERNAL_IP%
LOCAL_NET=%DEDICATED_NET%
EXTERNAL_DOMAIN=[указать доменное имя, по которому доступен сервер]
EXTERNAL_DOMAIN_HTTP_PROTO=(http|https)
EXTERNAL_DOMAIN_CONTACT=[указать свой контактный e-mail]
RHOMBUS_IP_ADDRESS_SEED=[указать доменное имя, по которому доступен сервер, а если отсутствует, то указать значение %EXTERNAL_IP%]

APP_VARIANT=dev
APP_INGRESS_VERSION=1.0.0-dev3
APP_CORE_STORAGE_VERSION=2.1.0-dev5
APP_CORE_APPLY_VERSION=2.1.0-dev3
APP_CORE_CONFIGURE_VERSION=2.1.0-dev4
APP_CORE_INSTANCES_VERSION=2.1.0-dev60
APP_HIPPO_PARROT_VERSION=2.1.0-dev5
APP_HIPPO_REPORTS_VERSION=2.0.0-dev3
APP_GIRAFFE_VERSION=2.0.0-dev33

PBX_SIC_VERSION=1.0.0-dev0
PBX_VERSION=1.0.0-dev6
PBX_CRON_VERSION=1.0.0-dev1
PBX_SIP_STATUS_SVC_VERSION=2.0.0-dev1
```

Версии приложений приведены на момент создание данной инструкции и могут отличаться от представленных в шаблоне. Следует использовать именно те версии, которые указаны в шаблоне `dist/.env`

Конфигурация сервиса применения конфигураций

Создаём файл `hippo-apply.env` со следующим содержимым:

```
DB_CONNECTION_STRING="pgsql://%PBX_CONFIG_DATABASE_USER%:%PBX_CONFIG_DATABASE_PASSWORD%@%PBX_CONFIG_DATABASE_HOST%:5432/"
```

Как пример:

```
DB_CONNECTION_STRING="pgsql://vats_pbx_user:password@psql.local:5432/vats_pbx"
```

Конфигурация сервиса построения отчётов

Создаём файл `hippo-reports.env` со следующим содержимым:

```
REPORT_DB_NAME=%CDR_DATABASE_NAME%
REPORT_DB_SERVER=%CDR_DATABASE_HOST%
REPORT_DB_USER=%CDR_DATABASE_USER%
REPORT_DB_PASSWORD=%CDR_DATABASE_PASSWORD%
```

Конфигурация сервиса управления пользователями (Keycloak)

Создаём файл `sso.env` в котором прописываем параметры БД, а также задаём инициализирующую учётную запись администратора сервиса управления пользователями:

```
KC_DB=postgres
KC_DB_URL=jdbc:postgresql://%KEYC_DATABASE_HOST%:5432/%KEYC_DATABASE_NAME%
KC_DB_USERNAME=%KEYC_DATABASE_USER%
KC_DB_PASSWORD=%KEYC_DATABASE_PASSWORD%

KEYCLOAK_ADMIN=admin
KEYCLOAK_ADMIN_PASSWORD=123456
KEYCLOAK_LOGLEVEL=DEBUG
```

Создание REALM в Keycloak

Создание базового realm можно произвести выполнив скрипт `install_realm.sh` из каталога `dist`. При этом случайным образом создадутся ключи для доступа к keycloak из запускаемых приложений.

Конфигурация сервиса управления ВТС

Создаём файл `hippo.env` со следующим содержимым:

```
DATABASE_URL="pgsql://%CONTROL_PLANE_DB_USER%:%CONTROL_PLANE_DB_PASSWORD%@%CONTROL_PLANE_DB_HOST%:5432/%CONTROL_PLANE_DB_NAME%"
CDR_DATABASE_URL="pgsql://%CDR_DATABASE_USER%:%CDR_DATABASE_PASSWORD%@%CDR_DATABASE_HOST%:5432/%CDR_DATABASE_NAME%"

PASSWORD_ALLOWED_CHARS="qwertyuiopasdfghjklzxcvbnmQWERTYUIOPASDFGHJKLZXCVBNM1234567890_"

PBX_DEFAULT_REGION=812
```

Значение параметра `PBX_DEFAULT_REGION` по умолчанию 812 код, что указывает на Санкт-Петербург. Данный параметр указывает номера телефонов с какими кодом будут считаться локальными, чтобы осуществлять ограничения направлений вызовов.

Некоторые sip клиенты не воспринимает некоторые спец-символы в качестве паролей, поэтому с помощью значения `PASSWORD_ALLOWED_CHARS` можно ограничить, какие символы допускаются.

Настройка Reverse-Proxy

По умолчанию, в качестве reverse-proxy для входящего трафика используется traefik, настроенный для получения ssl сертификатов с помощью Let's Encrypt.

Если у вас используется собственный сервис сертификации, обратитесь к документации traefik для внесения изменений в настройку.

Если же доступ по ssl не нужен, то можно удалить сервис reverse-proxy из файла `docker-compose.app.yml` и раскомментировать настройку `ports` в сервисе `ingress`. После этого не забудьте установить значение `EXTERNAL_DOMAIN_HTTP_PROTO=http` в файле `.env`

Осуществление запуска

Перед запуском проверить, что в каталоге с `docker-compose.yml` файлом присутствуют следующие файлы:

- `.env`
- `hippo-apply.env`

- hippo-reports.env
- hippo.env
- pbx_db.env
- sso.env

Для упрощения запуска поставляется скрипт *run.sh*, который проверит наличие необходимых файлов и запустит необходимые контейнеры

```
./run.sh
```

Для просмотра логов можно воспользоваться командой

```
docker compose logs -f
```

Выполняем миграции :

```
./migrate.sh
```

Загружаем минимально необходимые данные:

```
./seed.sh
```