

Установка на локальном сервере

Документ описывает процесс установки Webitel на собственном сервере в on-site варианте развертывания.

✔ Linux

Рекомендуем использовать [Debian Stretch](#) либо [Ubuntu Server](#). Только 64bit! Ниже приведенная инструкция подходит, в первую очередь, этим двум дистрибутивам.

Установка docker

Самый простой способ установки - это использование скрипта:

```
# apt-get install curl ntp
# curl -sSL https://get.docker.com/ | sh
# docker --version
Docker version 18.06.1-ce, build e68fc7a
```

Варианты установки на других дистрибутивах: <https://docs.docker.com/engine/installation/>

Установка docker-compose

Поскольку Webitel это комплекс из нескольких приложений, рекомендуем использовать [docker-compose](#). Установка:

```
# curl -L "https://github.com/docker/compose/releases/download/1.24.1/docker-compose-$(uname -s)-$(uname -m)" -
o /usr/local/bin/docker-compose
# chmod +x /usr/local/bin/docker-compose
# docker-compose --version
docker-compose version 1.24.1, build 1110ad01
```

Подготовка к запуску webitel

Для запуска Webitel Вам необходимо скачать подготовленные для `docker-compose` файлы, внести свои правки и запустить.

✔ Текущая стабильная версия [Webitel 3.11.3](#)

Получение файлов:

```
$ sudo gpasswd -a ${USER} docker
$ cd /opt
$ sudo mkdir /opt/webitel
$ sudo git clone https://github.com/webitel/orchestration.git
$ sudo chown -R ${USER}:docker /opt/webitel /opt/orchestration
$ cd /opt/orchestration
$ git checkout v3.11.3
```

В результате, мы получили копию подготовленных скриптов для запуска и управления контейнерами Webitel.

Конфигурационные файлы

В директории `/opt/orchestration/env` поставляются файлы конфигурации с расширением `*.example`. Вам необходимо скопировать эти файлы без расширения. Самый простой способ - выполнить команду:

```
$ cd /opt/orchestration/env
$ for f in *.example; do cp $f `basename $f .example`; done
```

Прежде, чем приступить к запуску, нам нужно изменить 2 значения в файле [env/environment](#):

- **WEBITEL_HOST** - необходимо указать полное FQDN имя или скрипт попытается определить IP адрес вашего сервера webitel автоматически.
- **WEBITEL_PASS** - пароль для пользователя root. Пароль может содержать символы латинского алфавита, цифры, а так же знаки: + - _ .

Параметры Linux:

```
$ sudo sh -c "cat /opt/orchestration/etc/sysctl.conf >> /etc/sysctl.conf"
$ sudo sysctl -p
```

Для Ubuntu необходимо дополнительно выполнить:

```
$ sudo apparmor_parser -R /etc/apparmor.d/usr.sbin.tcpdump
```

Добавить Webitel в автозагрузку:

```
$ sudo cp /opt/orchestration/etc/cron.d/webitel /etc/cron.d/
```



- [Установка собственного SSL сертификата](#)
- [Получение бесплатного Let's encrypt сертификата](#)

Запуск Webitel

Имея готовый файл с описанием контейнеров, мы можем скачать образы с Docker Hub на сервер и создать контейнеры.

Основные команды:

Команда	Описание
fs	Запустить fs_cli для подключения к консоли FreeSWITCH
backup	Выполняет резервное копирование базы данных и конфигурации webitel
-	не указывая ничего, запускаем webitel

Аргументы:

Аргумент	Описание
up	Создать либо пересоздать контейнеры. Для запуска контейнера в фоновом режиме, необходимо добавить ключ -d
stop	Остановить контейнеры
start	Запустить созданные раньше контейнеры
rm	Удалить контейнеры
pull	Скачать последнюю версию образов
ps	Вывести список контейнеров и их состояние
logs	Отобразить вывод логов из контейнеров

Запускаем webitel:

```
$ cd /opt/orchestration
```

```
./bin/bootstrap.sh pull
./bin/bootstrap.sh up -d
```

Проверяем, что все запустилось и работает:

```
./bin/bootstrap.sh ps
```

```
*****
webitel 3.11.0 software stack orchestration
*****

Webitel custom containers

-----
Name                Command                State                Ports
-----
acr                 ./acr -c ./config.json Up                   0.0.0.0:10030->10030/tcp
cdr                 ./cdr                  Up                   0.0.0.0:10024->10024/tcp
elasticsearch       /usr/local/bin/docker-entr ... Up                   0.0.0.0:9200->9200/tcp, 9300/tcp
elasticsearch2     /usr/local/bin/docker-entr ... Up                   0.0.0.0:9202->9200/tcp, 9300/tcp
engine              node server.js         Up                   172.17.0.1:10022->10022/tcp
freeswitch         /docker-entrypoint.sh free ... Up                   0.0.0.0:1024->5601/tcp
kibana              /entrypoint.sh kibana Up                   10.10.10.200:27017->27017/tcp
mongo               docker-entrypoint.sh --sto ... Up                   0.0.0.0:443->443/tcp, 0.0.0.0:80->80/tcp
nginx               /entrypoint.sh        Up                   10.10.10.200:5432->5432/tcp
postgres            /opt/cpm/bin/uid_postgres. ... Up                   10.10.10.200:5433->5432/tcp
postgres-rep       /opt/cpm/bin/uid_postgres. ... Up                   15671/tcp, 10.10.10.200:15672->15672/tcp, 25672/tcp, 4369/tcp, 5671/tcp,
rabbitmq            docker-entrypoint.sh rabbi ... Up                   10.10.10.200:5672->5672/tcp
storage            node server.js         Up                   172.17.0.1:10021->10021/tcp, 172.17.0.1:10023->10023/tcp, 172.17.0.1:10025->10025/tcp
wconsole           /docker-entrypoint.sh console Up                   0.0.0.0:10024->10024/tcp, 0.0.0.0:6873->6873/tcp
```

Что бы перезапустить полностью все контейнеры webitel, используйте последовательность:

```
./bin/reboot.sh
```

Наше приложение готово к работе.

Приложение	URL
Webitel WebClient	http://наш_домен/
WebSocket сервер и RESTful API	http://наш_домен/engine/
WebRTC подключение	http://наш_домен/verto/
Kibana	http://наш_домен/kibana/
Документация по API	http://наш_домен/docs/

Теперь можем переходить к разделу [Первичная настройка системы](#).



- [Webitel 3.11 upgrade notes](#)
- [Webitel 3.10 upgrade notes](#)
- [Webitel 3.9 upgrade notes](#)
- [Webitel 3.8 upgrade notes](#)
- [Webitel 3.7 upgrade notes](#)
- [Webitel 3.6 upgrade notes](#)
- [Webitel 3.5 upgrade notes](#)