

Инструкция для разворачивания front Abanking Classic

- Требования к инфраструктуре
 - Операционная система
 - Стороннее ПО
- Установка стороннего ПО на сервер
 - Установка Nginx на хост
 - Установка Docker
 - Установка docker-compose
- Распаковка контейнеров и загрузка образов
- Настройки Nginx
 - Настройки хостового Nginx
 - Применение новой конфигурации Nginx
- Запуск
 - Запуск приложения
- Проверка
 - Проверяем запущенные контейнеры

Требования к инфраструктуре

Операционная система

ОС	Версия
Debian	10
Ubuntu	20
ALT Linux	10
Astra Linux	1.7
RedOS	7.3.4

Стороннее ПО

Компонент	Версия	Ресурс
Docker	19.03.0	https://docs.docker.com/engine/install/
docker-compose	1.27.4	https://docs.docker.com/compose/install/linux/
Nginx	1.16.1	https://docs.nginx.com/nginx/admin-guide/installing-nginx/installing-nginx-open-source/

Обязательным требованием для запуска является наличие на хостовой машине следующего ПО:

- Docker
- docker-compose

Рабочая конфигурация предполагает наличие двух веб-серверов Nginx:

- nginx-1 - веб-сервер установленный на хостовой ОС;
- nginx-2 - контейнер с веб-сервером nginx

Инсталляция стороннего ПО на сервер

Установка Nginx на хост

Установка Docker

Установка docker-compose

docker compose версии 2.x.x и старше устанавливается вместе с пакетом docker engine.

Распаковка контейнеров и загрузка образов

На сервере, создать в корне каталог `/docker-containers`

Если с сервера нет доступа в интернет, для загрузки образов с Harbor, то поставка происходит через SFTP, для этого необходимо:

1. Загрузить архивы с SFTP на сервер, где будет размещен комплект поставки
2. Перейти в каталог с образами (каждый образ в архиве имеет вид `<SERVICE_NAME>.tar.gz`)
3. Выполнить загрузку следующей командой:

Распаковка и загрузка образов

```
sudo docker load -i <SERVICE_NAME>.tar.gz # Для загрузки одного образа
sudo ls -l *.tar.gz | xargs --no-run-if-empty -L 1 docker load -i # Для загрузки всех образов в каталоге
```

Проверить список загруженных образов можно командой:

Вывод списка образов

```
docker image ls
```

Перейти в каталог `/docker-containers`. Структура каталога, из комплекта поставки, соответствует примеру ниже.

Структура директорий

```
ibfl-front-release
├── nginx
│   ├── conf.d
│   ├── default.conf
│   └── log
├── settings
│   └── abanking-settings.prod.json #Настройки проекта
├── docker-compose.yml
└── .env
```

Настройки Nginx

Настройки хостового Nginx

Добавить проксирование с веб-сервера реверс-прокси, установленный непосредственно на хост с сервисами.

В примере ниже - если на хосте установлен Nginx, то конфигурация проксировала запросы на Nginx в контейнере).

Директиву `server_name` заменить на свой домен.

Пример конфигурации хостового Nginx

```
server {
    # публичные домены
    server_name
        <some_name> # домены

    listen 80;

    location / {

        client_max_body_size 30m;

        proxy_set_header Host $host;
        proxy_set_header X-Forwarded-Proto $scheme;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;

        # блок настроек необходимый для проксирования websocket запросов
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection $http_connection;
        proxy_cache_bypass $http_upgrade;
        proxy_http_version 1.1;

        proxy_pass http://127.0.0.1:10000/;

    }
    #для блокировки swagger'a, интернал рестов, дебаг рестов

    if ($request_uri ~* (swagger|internal|debug)) {
        set $block_me_now A;
    }

    if ($remote_addr != xx.xx.xx.xx) { # добавить разрешенный адрес
        set $block_me_now "${block_me_now}B";
    }

    if ($block_me_now = AB) {
        return 403;
        break;
    }

    location ~ .*(/internal/.*) {
        deny all;
    }

    location ~ .*(/debug/.*) {
        deny all;
    }

    location ~ .*(/swagger/.*) {
        deny all;
    }

    location ~ .*(/document-api/public/.*) {
        deny all;
    }
}
```

Применение новой конфигурации Nginx

После добавления файла конфигурации требуется проверить его на синтаксические ошибки, если они отсутствуют - выполнить перезагрузку хвостового Nginx для их активации:

Применение конфигурации

```
sudo nginx -t          # Проверка синтаксиса файла конфигурации
sudo nginx -s reload   # Перезапуск Nginx
```

В файле `/docker-containers/ibfl-front-release/docker-compose.yml` указан список контейнеров с конфигурациями запуска. В нём можно использовать переменные для тегов.

Пример указания переменной тега

```
application-browser:
  image: artsoftе/application:${VER}
  networks:
    - network
  volumes:
    - "/application/settings/:/usr/share/nginx/html/settings"
  restart: unless-stopped
```

В файле `/docker-containers/ibfl-front-release/.env` можно задать переменную тегов.

Пример указания тегов версий

```
VER=app-release-145
```

Запуск

Запуск приложения

Запустить образы приложений из директории `/docker-containers/ibfl-front-release`

Запуск микросервисов

```
cd /docker-compose/services
docker-compose up -d
```

Проверка

Проверяем запущенные контейнеры

Проверить, что контейнеры не в рестарте, имеют статус `Up` больше 20 секунд. Команда выполняется без привязки к каталогу:

Вывод информации о запущенных контейнерах

```
docker ps
```