

Инструкция по установке экземпляра программного обеспечения Система управления образовательным процессом для поддержки курсов и акселераторов «Distant Global»

Платформа представляет собой два приложения, распространяющихся в виде docker-контейнеров.

Приложения могут быть запущены в одно-серверной конфигурации, описанной ниже. База данных может использоваться как отдельно стоящая, так и развёрнутая на сервере приложений. Приложения устанавливаются путем запуска docker-контейнеров с использованием docker-compose.yml файла.

Состав приложений платформы:

- **dg-api** — бэкэнд программного продукта;
- **dg-front** — фронт-энд программного продукта;
- **mongo** — key-value БД MongoDB лицензия GNU AGPL v3.0;
(<https://www.mongodb.com/legal/licensing/community-edition>);

Экземпляр программного продукта доступен по ссылке, предоставленной заказчику в личном кабинете, либо по запросу на электронную почту поддержки support@distant.global

1. Подготовка окружения

Обязательным предусловием для установки серверной платформы является установка следующих пакетов:

- docker, начиная с версии 20.10
- docker-compose, начиная с версии 2.21

Детали их установки описаны в документах

<https://docs.docker.com/engine/install/debian/> (в подразделе "convenience script") и <https://docs.docker.com/compose/install/linux/> (установка в виде plugin)

Дополнительная документация по установке docker-compose:

<https://github.com/docker/docker-install/>

В большинстве случаев для установки docker и docker-compose следует запустить команды:

```
$ curl -fsSL https://get.docker.com -o get-docker.sh
$ sudo sh ./get-docker.sh
$ apt install docker-compose-plugin
```

В случае отсутствия программы `curl`, для скачивания установочного скрипта можно воспользоваться `wget`:

```
$ wget https://get.docker.com -qO get-docker.sh
```

Запуск контейнеров Платформы рекомендуется производить из-под непривилегированного пользователя (например, с именем **user**). Пользователь должен быть участником группы **docker**.

Команда для добавления пользователя в систему и добавления пользователя в группу:

```
$ sudo useradd -m -s /bin/bash user
$ sudo usermod -aG docker user
```

В дальнейшем запуск контейнеров необходимо производить из-под выбранного аккаунта **user**.

2. Установка программного продукта

Для установки программного продукта требуется выполнить следующие шаги:

- получить установочные файлы
- сконфигурировать переменные в файле окружения `.env`
- настроить `nginx` для домена приложения
- аутентифицироваться в `repo.distant.global` средствами `docker`
- запустить `docker compose`

2.1 Получение установочных файлов

В домашней директории пользователя **user** выполните следующие команды для получения установочных файлов:

```
$ mkdir -p ~/distant-global; cd ~/distant-global
$ wget https://distant.global/latest/dg.tar.gz
$ tar xfa dg.tar.gz
```

Набор файлов, находящихся в архиве:

- `.env` - файл с переменными окружения
- `docker-compose.yml` - манифест запускаемых контейнеров
- `nginx.default.template` - шаблон конфигурации `nginx` для сайта

2.2 Подготовка переменных

Необходимо установить требуемые значения для переменных в файле `.env` - данные в этом файле будут использоваться при запуске контейнеров.

<code>SERVERNAME=testsite.distant.global</code>	доменное имя платформы
<code>IMAGE_API=repo.distant.global/dg-api:6b091d73</code>	имя образа для приложения <code>dg-api</code>
<code>PORT_API=5000</code>	порт приложения <code>dg-api</code> по-умолчанию
<code>IMAGE_FRONT=repo.distant.global/dg-front:dfa0ce47</code>	имя образа для приложения <code>dg-front</code>
<code>PORT_FRONT=4000</code>	порт приложения <code>dg-front</code> по-умолчанию
<code>YANDEX_FORM_CODE_NON_AUTH=</code>	форма технической поддержки для неавторизованных пользователей
<code>YANDEX_FORM_CODE_AUTH=</code>	форма технической поддержки для авторизованных пользователей
<code>KINESCOPE_PROJECT_ID=</code>	идентификатор проекта в CDN Kinescope
<code>KINESCOPE_API_KEY=</code>	ключ проекта в CDN Kinescope
<code>YANDEX_CAPTCHA_KEY=</code>	Капча яндекс-формы поддержки обращений неавторизованных пользователей
<code>DADATA_KEY=</code>	ключ для справочников сервиса Dadata
<code>IMAGE_MONGO=networkweb/mongodb-no-avx:latest</code>	имя образа базы данных
<code>DB_VOLUME=/opt/dbdata</code>	директория для файлов базы данных

2.3 Установить поставляемый конфиг nginx для сайта

Действия по данному пункту следует проводить либо при помощи `sudo`, либо от пользователя `root`.

Файл `nginx.default.template` скопировать в `nginx.default.conf` и установить его к остальным конфигам в `/etc/nginx`.

Местоположение папки с конфигами разнится от ОС к ОС, но, например, для Ubuntu этой папкой является

`/etc/nginx/sites-available`.

```
server {
    server_name  distant-global.example.com ; ## SERVERNAME
    resolver 8.8.8.8;

    listen      80;
    listen  [::]:80;

    access_log
/var/log/nginx/distant-global.example.com.access.log ;
    error_log   /var/log/nginx/distant-global.example.com.error.log
;

    client_max_body_size 30M;

    location / {
        proxy_pass  http://127.0.0.1:4000; ## PORT_FRONT
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection "Upgrade";
        proxy_set_header Host $host;
        proxy_cache_bypass $http_upgrade;

        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For
$proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;
####      add_header 'Access-Control-Allow-Origin' '*';
    }

    location ~* ^/(api/v1|api-docs|socket\.io)\.+ {
        proxy_pass  http://127.0.0.1:5000; ## PORT_API
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Host $host;
        proxy_set_header Connection "Upgrade";
        proxy_cache_bypass $http_upgrade;
    }
}
```

(для ubuntu) После копирования создайте символическую ссылку:

```
$ sudo ln -s /etc/nginx/sites-available/nginx.default.conf  
/etc/nginx/sites-enabled/nginx.default.conf
```

Проверьте правильность установок запуском

```
$ sudo nginx -T
```

Перезапустите nginx (в ubuntu: `$ sudo /etc/init.d/nginx restart`)

2.4 Получение образов контейнеров Приложения

Для целей тестирования некоторые версии образов `dg-api` и `dg-front` могут быть загружены посредством прямого скачивания следующими командами:

```
$ wget https://distant.global/latest/dg-api.tar.gz -O - |  
docker image load  
$ wget https://distant.global/latest/dg-front.tar.gz -O - |  
docker image load
```

Для получения рабочих версий образов необходимо авторизоваться следующей командой из под пользователя **user** :

```
$ docker login --username USERNAME --password PASSWORD  
repo.distant.global
```

Учётные данные `USERNAME` и `PASSWORD` можно получить в технической поддержке.

2.5 Запуск контейнеров

Перейти в директорию и выполнить запуск контейнеров командой:

```
$ cd ~/distant-global  
$ docker compose up -d
```

В процессе запуска будут получены образы `dg-api`, `dg-front`, `mongo` и запущены соответствующие контейнеры.

Продиагностировать запуск контейнеров можно командами:

```
$ docker ps -a  
$ docker logs dg-api  
$ docker logs dg-front
```