



Корпоративный сервис для
защищенного обмена данными

Руководство по установке MFlash в контейнерах Docker

Установка Docker для Astra linux, RedOs

Docker устанавливается из официального репозитория командой и далее включается для автозагрузки после перезапуска системы:

1. `sudo apt install docker-ce`
2. `sudo systemctl enable docker`
3. `sudo systemctl start docker`

Если требуется установить наиболее свежую версию, необходимо выполнить следующие действия:

Добавим новый источник пакета, ключ GPG от Docker, чтобы гарантировать загрузку рабочих файлов, а затем установим пакет.

Первым делом обновите существующий список пакетов:

1. `sudo apt update`

Затем установите несколько необходимых пакетов, которые позволяют apt использовать пакеты через HTTPS:

2. `sudo apt install apt-transport-https ca-certificates curl gnupg2 software-properties-common`

Добавьте ключ GPG для официального репозитория Docker в вашу систему:

3. `curl -fsSL https://download.docker.com/linux/debian/gpg | sudo apt-key add -`

Добавьте репозиторий Docker в источники АРТ:

4. `sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/debian $(lsb_release -cs) stable"`

Потом обновите базу данных пакетов и добавьте в нее пакеты Docker из недавно добавленного репозитория:

1. `sudo apt update`

Установите Docker:

1. `sudo apt install docker-ce`

Docker должен быть установлен, демон-процесс запущен, а для процесса активирован запуск при загрузке. Проверьте, что он запущен:

1. `sudo systemctl status docker`

Развёртывание системы в контейнерах:

Рабочая директория: /opt/mflash

Все контейнеры должны быть загружены в систему перед запуском сервисов. Контейнеры высылаются либо в виде прямой ссылки на архив, либо загружаются из нашего репозитория `hub.msoftgroup.ru:5000` используя команду:

```
docker login hub.msoftgroup.ru:5000
```

Username: mflash

Password: aVshENNLHy9ea5n6

После получения образов системы необходимо выполнить команду

```
docker load < путь_и_имя_архива.tar
```

Проверить, что образ загрузился

```
Docker images
```

Имена образов могут быть вида `0e5574283393`, их необходимо как пример переименовать командой, чтобы далее прописать в файле конфигурации сервиса данное имя

```
Docker tag 0e5574283393 mf-app:7.1.84
```

Список сервисов:

- mflash-service
- mflash-worker
- nginx-docker
- mysql-docker
- mflash-web

Порядок запуска:

- MySQL
- WEB
- Nginx
- DATA

Если на хостовой системе стоит **podman** а не **docker**, то нужно установить пакет: **podman-docker**

```
dnf install podman-docker
```

Во всех *.service изменить поля After и Requires на: **podman.service**

Для компонента WEB:

Необходимо выполнить команду:

```
docker volume create web_spase
```

Далее создать файл `/etc/systemd/system/mflash-web.service` со следующим содержимым:

```
[Unit]
Description=MFlash Web
After=docker.service
Requires=docker.service

[Service]
TimeoutStartSec=0
Restart=always
ExecStartPre=-/usr/bin/docker exec %n stop
ExecStartPre=-/usr/bin/docker rm %n
ExecStart=/usr/bin/docker run --rm --name %n \
    -v /opt/mflash/logs:/var/log/ \
    -v web_spase:/srv/mflash \
    -e MF_DB_IP={MYSQL IP} \
    -e MF_DB_USER=root \
    -e MF_DB_PASSWORD=mflash01 \
    -p 9000:9000 \
    mf-app:7.1.84

[Install]
WantedBy=default.target
```

{MYSQL IP} - Заменить на IP адрес MySQL сервера (Example: 10.1.1.1)

Для компонента Nginx (*Должен стоять на том же сервере что и WEB*):

Создать файл `/etc/systemd/system/nginx-docker.service` со следующим содержимым:

```
[Unit]
Description=Nginx
After=docker.service
Requires=docker.service

[Service]
TimeoutStartSec=0
Restart=always
ExecStartPre=-/usr/bin/docker exec %n stop
ExecStartPre=-/usr/bin/docker rm %n
ExecStart=/usr/bin/docker run --rm --name %n \
    -v /opt/mflash/logs:/var/log/nginx \
    -v web_spase:/srv/mflash \
    -v /opt/mflash/mf_conf/mflash.conf:/etc/nginx/conf.d/mflash.conf \
    -p 80:80 \
    -p 443:443 \
    nginx:latest

[Install]
WantedBy=default.target
```

В mf_conf/mflash.conf:

fastcgi_pass [app_web:9000](#); Изменить на IP сервера

Для компонента DATA:

Создать файл /etc/systemd/system/mflash-service.service со следующим содержимым:

```
[Unit]
Description=MFlash Service
After=docker.service
Requires=docker.service

[Service]
TimeoutStartSec=0
Restart=always
ExecStartPre=-/usr/bin/docker exec %n stop
ExecStartPre=-/usr/bin/docker rm %n
ExecStart=/usr/bin/docker run --rm --name %n \
    -v /opt/mflash/smw:/srv/mflash/smw \
    -w /opt/mflash/data/service/ \
    -e MF_DB_IP={MYSQL IP} \
    -e MF_DB_USER=root \
    -e MF_DB_PASSWORD=mflash01 \
    -p 11000:11000 \
    mf-svc:7.1.22 \
    dotnet mfservice.dll

[Install]
WantedBy=default.target
```

{MYSQL IP} - Заменить на IP адрес MySQL сервера (Example: 10.1.1.1)

Создать файл /etc/systemd/system/mflash-worker.service со следующим содержимым:

```
[Unit]
Description=MFlash Worker
After=docker.service
Requires=docker.service

[Service]
TimeoutStartSec=0
Restart=always
ExecStartPre=-/usr/bin/docker exec %n stop
ExecStartPre=-/usr/bin/docker rm %n
ExecStart=/usr/bin/docker run --rm --name %n \
    -v /opt/mflash/smw:/srv/mflash/smw \
    -w /opt/mflash/data/worker/ \
    -e MF_DB_IP={MYSQL IP} \
    -e MF_DB_USER=root \
    -e MF_DB_PASSWORD=mflash01 \
    mf-svc:7.1.23 \
    dotnet mfwork.dll

[Install]
WantedBy=default.target
```

{MYSQL IP} - Заменить на IP адрес MySQL сервера (Example: 10.1.1.1)

Для компонента MySQL:

Создать папки:

```
mkdir -p /opt/mflash/logs
mkdir -p /opt/mflash/db
chown 1000:50 /opt/mflash/db
mkdir -p /opt/mflash/mf_conf
```

Далее создать файл `/etc/systemd/system/mysql-docker.service` со следующим содержимым:

```
[Unit]
Description=MySQL 5.7
After=docker.service
Requires=docker.service

[Service]
TimeoutStartSec=0
Restart=always
ExecStartPre=-/usr/bin/docker exec %n stop
ExecStartPre=-/usr/bin/docker rm %n
ExecStart=/usr/bin/docker run --rm --name %n \
  -v /opt/mflash/db:/var/lib/mysql \
  -v /opt/mflash/logs:/var/log/ \
  -v /opt/mflash/mf_conf/mflash.cnf:/etc/mysql/conf.d/mflash.cnf \
  -e MYSQL_ROOT_PASSWORD=mflash01 \
  -p 3306:3306 \
  --user 1000:50 \
  mysql:5.7

[Install]
WantedBy=default.target
```

Директория `/opt/mflash/mf_conf`:

Создать файл `mflash.conf`: со следующим содержимым:

```
server {
    listen 80;
    server_name astra;
    client_max_body_size 2g;

    location ~ /(admin|api|client|mflash|share)/ {
        root /srv/mflash/web/;
        index index.php index.html index.htm;

        location ~ ^/(admin|api|client|mflash|share)/(.*)\.php$ {
            fastcgi_read_timeout 6400;
            fastcgi_pass 127.0.0.1:9000;
            fastcgi_index index.php;
            fastcgi_param SCRIPT_FILENAME
/srv/mflash/web$fastcgi_script_name;
            include fastcgi_params;
        }
    }

    location ~ /api_v2(.*) {
```



```
systemctl start mflash-worker
```

Поставить сервисы в автозапуск:

```
systemctl enable mflash-service
```

Для добавления сертификатов их необходимо прописать в конфиге nginx, в файле mflash.conf

```
server{
listen 80;
return 301 https://адрес_системы;
}

server {
    listen 443 default_server ssl;
    ssl_certificate /etc/nginx/certs/cert_name.crt;
    ssl_certificate_key /etc/nginx/certs/cert_key_name.key ;
    ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
    ssl_ciphers HIGH:!aNULL:!MD5;
    server_name pfo.transneft.ru;
    client_max_body_size 2g;
}
```

Далее добавить параметр в конфигурацию сервиса nginx-docker.service:

```
-v /opt/mflash/mf_conf/:/etc/nginx/certs \
```


Обновление системы

При выпуске новой версии необходимо выполнить следующие пункты:

После получения образов системы необходимо выполнить команду

```
docker load < путь_и_имя_архива.tar
```

Проверить, что образ загрузился

```
Docker images
```

Имена образов могут быть вида 0e5574283393, их необходимо как пример переименовать командой, чтобы далее прописать в файле конфигурации сервиса данное имя

```
Docker tag 0e5574283393 mf-app:7.1.84
```

Затем в файлах `service` изменить строку на нужную версию, выполнить команду `systemctl daemon-reload` и перезапустить службы, отвечающие за обновлённые компоненты.