

Обновление Loginom в системе контейнеризации

Перед выполнением обновления рекомендуется сделать резервную копию рабочей директории, чтобы в случае неудачного обновления можно было вернуться к версии, установленной ранее. Если Loginom был установлен с путями по умолчанию, то для этого необходимо выполнить команду:

```
cp -r /var/opt/loginom-docker/ /var/opt/loginom-docker-backup
```

Подготовка дистрибутива

Перед началом обновления выполняется подготовка дистрибутива, для этого необходимо распаковать архив с дистрибутивом, выполнив команду:

```
tar -xf <архив_с_дистрибутивом>.tar
```

В результате образуется директория, внутри которой располагается архив с установочным скриптом. Архив `loginom-install-docker*.tgz` необходимо распаковать и перейти в образовавшуюся директорию:

```
1 cd <директория_с_дистрибутивом>
2 tar -xf <архив_с_установочным_скриптом>.tgz
3 cd <директория_с_установочным_скриптом>
```

Дополнительные компоненты

Если требуется установка дополнительных компонентов, таких как Python, ODBC-драйверы и прочее, то необходимо указать это в файле-сценарии сборки образа `server/Dockerfile`. Отредактировать файл можно, например, с помощью команды:

```
nano ./server/Dockerfile
```

▼ Пример Dockerfile

```
1 FROM docker.io/debian:trixie-slim
2 #FROM docker.io/debian:bookworm-slim
3 #FROM docker.io/debian:bullseye-slim
4 #FROM docker.io/ubuntu:22.04
5 #FROM docker.io/ubuntu:24.04
6 #FROM registry.astralinux.ru/library/astra/ubi18:1.8
7
8 LABEL "org.opencontainers.image.vendor"="Loginom Company"
9 LABEL "org.opencontainers.image.title"="Loginom Server"
10
11 ARG DEBIAN_FRONTEND=noninteractive
12 ARG ACCEPT_EULA=Y
13
14 COPY server/ /app/
15 COPY libs/ /app/
16
17 RUN apt-get update && apt-get install -y --no-install-recommends \
18     tzdata \
19     libpcre2-8-0 \
20     zlib1g \
21     '^libc[u678][0-9]$\ ' \
22     librdkafka1 \
23     libmariadb3 \
24     libsqlite3-0 \
25     #unixodbc \
26     #tdsodbc \
27     libtomcrypt1 \
28     libtommath1 && \
29     apt-get clean && rm -rf /var/lib/apt/lists/* && \
30     mkdir -p /app/firebird/lib && \
31     ln -snf /app/firebird/lib/libfbclient.so.5.0.2
/app/firebird/lib/libfbclient.so.2 && \
32     ln -snf /app/firebird/lib/libfbclient.so.2
/app/firebird/lib/libfbclient.so && \
33     ln -snf /app/firebird/lib/libfbclient.so /usr/lib/x86_64-linux-
gnu/libfbclient.so
34
35 # Installing the package required to add certificates into container
{#installing-the-package-required-to-add-certificates-into-container}
```

```
36 #RUN apt-get update && apt-get install -y --no-install-recommends ca-
certificates && \
37 # apt-get clean && rm -rf /var/lib/apt/lists/* {#apt-get-clean--rm--
rf-varlibaptlists}
38
39 # Installing Microsoft ODBC Driver 18 for SQL Server {#installing-
microsoft-odbc-driver-18-for-sql-server}
40 # The msodbcsql18_18.4.1.1-1_amd64.deb file should be located in the
'server/libs' directory of the distribution {#the-msodbcsql1818411-
1amd64deb-file-should-be-located-in-the-serverlibs-directory-of-the-
distribution}
41 #RUN apt-get update && apt-get install -y --no-install-recommends \
42 # libgssapi-krb5-2 \ {#libgssapi-krb5-2-}
43 # /app/msodbcsql18_18.4.1.1-1_amd64.deb \ {#appmsodbcsql1818411-
1amd64deb-}
44 # locales && \ {#locales--}
45 # localedef -i ru_RU -f UTF-8 ru_RU.UTF-8 && \ {#localedef--i-ruru--
f-utf-8-ruruutf-8--}
46 # rm -f /app/msodbcsql18_18.4.1.1-1_amd64.deb && \ {#rm--f-
appmsodbcsql1818411-1amd64deb--}
47 # apt-get clean && rm -rf /var/lib/apt/lists/* {#apt-get-clean--rm--
rf-varlibaptlists}
48
49 # Installing Python and some of its libraries {#installing-python-and-
some-of-its-libraries}
50 #RUN apt-get update && apt-get install -y --no-install-recommends
python3 \
51 # python3-pandas python3-numpy python3-sklearn && \ {#python3-pandas-
python3-numpy-python3-sklearn--}
52 # apt-get clean && rm -rf /var/lib/apt/lists/* {#apt-get-clean--rm--
rf-varlibaptlists}
53
54 ENV LANG=ru_RU.UTF-8
55 ENV LD_PRELOAD=/app/libtcmalloc_minimal.so.4.4.5
56 ENV TCMALLOC_AGGRESSIVE_DECOMMIT=true
57 ENTRYPOINT ["/app/loginomd", "--run", "-workdir=/workdir"]
```

Остановка контейнеров

Перед началом обновления необходимо остановить работающие контейнеры. Для этого нужно выполнить команду:

```
sudo docker compose -f /var/opt/loginom-docker/docker-compose.yml down
```

Обновление образов

Для выполнения обновления необходимо запустить установочный скрипт с параметром `update` с помощью команды:

```
sudo bash setup.sh --update
```

Запуск контейнеров

После обновления образов необходимо запустить контейнеры, выполнив команду:

```
sudo docker compose -f /var/opt/loginom-docker/docker-compose.yml up -d
```

Чтобы убедиться, что контейнеры запущены и работают, можно выполнить команду:

```
sudo docker ps -a
```

Должен отобразиться список всех контейнеров — как работающих, так и нет.

Примечание: В данной инструкции представлены команды для Loginom, установленного с использованием **Docker**. В случае с **Podman** инструкция по обновлению аналогична за исключением некоторых особенностей, описанных в статье [Установка Loginom в систему контейнеризации Podman или Docker](#).