

Установка «СУБД-КС Docker» на Альт Сервер 10 (build 10.1)

Платформа для развертывания, управления и удаления экземпляра СУБД-КС 15.2, реализованного в виде контейнера для работы в средах с поддержкой контейнеризации Docker или Podman.

Требует для работы: Docker-CE (Docker, Inc. лицензия Apache 2.0) или Podman (лицензия Apache 2.0), платформа linux/amd64.

Характеристики: Позволяет устанавливать один экземпляр СУБД-КС по указанному порту. Диапазон используемых портов: 5432-5444.

1. Подключиться к Серверу по ssh, авторизоваться.

2. Необходимо получить root-права:

```
su -
```

3. Проверьте, установлен ли «Docker-CE». Для этого введите:

```
docker -v
```

Если отобразится информация о версии установленного «Docker-CE» как показано на рисунке, то пропустите шаги № 4 и 5!

```
[root@host-142 ~]# docker -v
Docker version 20.10.21, build baeda1f
[root@host-142 ~]#
```

4. Скачайте и распакуйте архив «Docker-CE». Для этого введите:

```
wget -O /opt/dks-docker-ce-20.10.21.tar.gz https://keysystems.ru/files/web/Linux/docker/dks-docker-ce-20.10.21.tar.gz --no-check-certificate
```

```
tar -C /opt/ -xvf /opt/dks-docker-ce-20.10.21.tar.gz
```

5. Установите «Docker-CE». Для этого выполните команду:

```
bash /opt/dks-docker-ce-20.10.21/install.sh
```

При успешной установке «Docker-CE» получите следующие сообщения:

```
[root@host-142 ~]# bash /opt/dks-docker-ce-20.10.21/install.sh
~
~ Установка Docker-CE 20.10.21 ~
~
[ OK ] Проверка/Создание группы docker.
[ OK ] Распаковка архива docker-20.10.21.tgz.
[ OK ] Копирование бинарных файлов в каталог /usr/bin/.
[ OK ] Создание файла сервиса docker.service.
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /lib/systemd/system/docker.service.
[ OK ] Запуск сервиса docker.service.
```

6. Скачайте и распакуйте архив «СУБД-КС Docker»:

```
wget -O /opt/dks-pgsqlks-15.2_d11.tar.gz https://keysystems.ru/files/web/Linux/pgsqlks/15.2/dks-pgsqlks-15.2_d11.tar.gz --no-check-certificate
```

```
tar -C /opt/ -xvf /opt/dks-pgsqlks-15.2_d11.tar.gz
```

7. Изучите параметры установки:

```
bash /opt/dks-pgsqlks-15.2/install.sh --help
```

```
[root@host-142 ~]# bash /opt/dks-pgsqqlks-15.2/install.sh --help
```

параметры	по умолчанию	описание
-p --port	5434	порт СУБД
-pwd --password	1	пароль, который присваивается суперпользователям СУБД dbo и postgres
-l --locale	ru_RU.UTF-8	локаль для инициализации кластера СУБД
-shm --shm_size	256	размер shared memory при создании контейнера в MB
-cs --contsoft	docker	ПО контейнеризации (docker или podman)

Детальное описание работы с «СУБД-КC Docker» содержится в файле «readme.txt». Для его просмотра выполните команду:

```
cat /opt/dks-pgsqqlks-15.2/readme.txt
```

8. Установите пакет dmidecode.

```
apt-get update
```

```
apt-get install dmidecode
```

```
[root@host-142 ~]# apt-get install dmidecode
Чтение списков пакетов ... Завершено
Построение дерева зависимостей ... Завершено
Следующие НОВЫЕ пакеты будут установлены:
 dmidecode
0 будет обновлено, 1 новых установлено, 0 пакетов будет удалено и 177 не будет обновлено.
Необходимо получить 59,4кВ архивов.
После распаковки потребуется дополнительно 193кВ дискового пространства.
Получено: 1 http://ftp.altlinux.org p10/branch/x86_64/classic dmidecode 3.3-alt1:sisyphus+278447.100.1.1@1626101023 [59,4кВ]
Получено 59,4кВ за 0s (733кВ/с).
Совершаем изменения ...
Подготовка ...
Обновление / установка ...
1: dmidecode-3.3-alt1
Завершено.
```

9. Установите «СУБД-КC Docker» порт 5432(пример) с паролем «123» (пример) на пользователя dbo и postgres в Docker. Сначала определите установленные локали.

Для этого выполните команду:

```
locale -a | grep ru
```

Пример вывода:

```
[root@host-142 ~]# locale -a | grep ru
ru_RU
ru_RU.cp1251
ru_RU.ibm866
ru_RU.iso88595
ru_RU.koi8r
ru_RU.utf8
ru_UA
ru_UA.cp1251
ru_UA.koi8u
ru_UA.utf8
```

Установите СУБД:

```
bash /opt/dks-pgsqqlks-15.2/install.sh -p 5432 -pwd 123 -cs docker -l ru_RU.utf8
```

```
[root@host-142 ~]# bash /opt/dks-pgsqlks-15.2/install.sh -p 5432 -pwd 123 -cs docker -l ru_RU.utf8

~ Установка контейнера СУБД-КС 15.0
~ порт : 5432
~ контейнеризация : docker

[ OK ] Проверка доступности порта 5432 по "сервер СУБД".
[ OK ] Проверка корректности значения "256" параметра контейнера "Shared Memory".
[ OK ] Проверка работы сервиса по "Docker" (docker.service).
[ OK ] Создание пользователя ks-postgres.
[ OK ] Загрузка эталонного локального образа pgsqlks-15.2 из архива ksdeb11_pgsqlks-15.2_image.tar.
[ OK ] Построение конечного образа pgsqlks-15.2_5432 на базе эталонного.
[ OK ] Создание контейнера pgsqlks-15.2_5432 и запуск в docker.
[ OK ] Запуск контейнера pgsqlks-15.2_5432.
[ OK ] Удаление эталонного локального образа pgsqlks-15.2.
[ OK ] Переключение локали контейнера на "ru_RU.utf8".
[ OK ] Кластер СУБД pgsqlks-15.2 не существует, создаем ...
[ OK ] Создание кластера СУБД pgsqlks-15.2.
[ OK ] Запуск сервера СУБД pgsqlks-15.2:5432.
[ OK ] Установка пароля пользователю postgres.
[ OK ] Создание\коррекция пользователя dbo.
[ OK ] Создание служебных процедур в БД postgres.
[ OK ] Установка пароля пользователю dbo.
[ OK ] Сервер СУБД pgsqlks-15.2 доступен в сети по порту 5432.
[ OK ] Остановка контейнера pgsqlks-15.2_5432.
[ OK ] Created symlink /etc/systemd/system/multi-user.target.wants/pgsqlks-15.2_5432.service → /lib/systemd/system/pgsqlks-15.2_5432.service.
[ OK ] Запуск сервиса pgsqlks-15.2_5432.service.
```

10. Проверьте работоспособность СУБД-КС. Для этого выполните запрос.

```
bash /opt/pgsqlks-15.2/utility/5432/psql.sh -h 127.0.0.1 -c "select datname, pg_size_pretty(pg_database_size(datname)) from pg_database;"
```

После успешного выполнения запроса увидим следующее окно:

```
[root@host-142 ~]# bash /opt/pgsqlks-15.2/utility/5432/psql.sh -h 127.0.0.1 -c "select datname, pg_size_pretty(pg_database_size(datname)) from pg_database;"

 datname | pg_size_pretty
-----+-----
 postgres | 7293 kB
 template1 | 7065 kB
 template0 | 7065 kB
(3 rows)

[ OK ] : Docker_sql_command_completed
```

Установка «WEB-Сервер-КС» на Альт Сервер 10 (build 10.1)

Платформа для развертывания, управления и удаления http-сервера, реализованного в виде контейнера для работы в средах с поддержкой контейнеризации Docker или Podman.

Требует для работы: Docker-CE (Docker, Inc. лицензия Apache 2.0) или Podman (лицензия Apache 2.0), платформа linux/amd64.

Характеристики: Позволяет устанавливать один контейнер с http-сервером Apache или Nginx по указанному порту, внутри которого возможна установка нескольких .NET Core приложений идентифицируемых назначенным виртуальным каталогом. Диапазон используемых портов: 5444-5493 и дополнительно 443 порт (по умолчанию). В поставляемом образе контейнера используются компоненты:

- http сервер Apache 2.4.25/2.4.53 (лицензия Apache 2.0)
- http сервер Nginx 1.18.0 (лицензия BSD из 2 пунктов)
- Runtime библиотека .NET Core 3.1.24 (лицензия MIT)
- prometheus 2.31.1 (лицензия Apache 2.0)

1. Подключиться к Серверу по ssh, авторизоваться.

2. Необходимо получить root-права:

```
su -
```

3. Проверьте, установлен ли «Docker-CE». Для этого введите:

```
docker -v
```

Если отобразится информация о версии установленного «Docker-CE» как показано на рисунке, то пропустите шаги № 4 и 5!

```
[root@host-142 ~]# docker -v
Docker version 20.10.21, build baeda1f
[root@host-142 ~]#
```

4. Скачайте и распакуйте архив «Docker-CE». Для этого введите:

```
wget -O /opt/dks-docker-ce-20.10.21.tar.gz https://keysystems.ru/files/web/Linux/docker/dks-docker-ce-20.10.21.tar.gz --no-check-certificate
```

```
tar -C /opt/ -xvf /opt/dks-docker-ce-20.10.21.tar.gz
```

5. Установите «Docker-CE». Для этого выполните команду:

```
bash /opt/dks-docker-ce-20.10.21/install.sh
```

При успешной установке «Docker-CE» получите следующие сообщения:

```
[root@host-142 ~]# bash /opt/dks-docker-ce-20.10.21/install.sh
~ ~ ~ ~ ~
~ Установка Docker-CE 20.10.21 ~
~ ~ ~ ~ ~
[ OK ] Проверка/Создание группы docker.
[ OK ] Распаковка архива docker-20.10.21.tgz.
[ OK ] Копирование бинарных файлов в каталог /usr/bin/.
[ OK ] Создание файла сервиса docker.service.
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service -> /lib/systemd/system/docker.service.
[ OK ] Запуск сервиса docker.service.
```

6. Скачайте и распакуйте архив «WEB-Сервер-КС»:

```
wget -O /opt/dks-wsks-3.1.24_d11.tar.gz https://keysystems.ru/files/web/Linux/wsks/3.1.24/dks-wsks-3.1.24_d11.tar.gz --no-check-certificate
```

```
tar -C /opt/ -xvf /opt/dks-wsks-3.1.24_d11.tar.gz
```

7. Ознакомьтесь при необходимости с описанием «WEB-Сервер-КС».

```
cat /opt/dks-wsks-3.1.24/readme.txt
```

8. Установите контейнер Docker+Nginx (порт web-сервера 443, тип nginx, порт Prometheus 9091).

```
/opt/dks-wsks-3.1.24/install.sh -ws "nginx" -p "443" -cs "docker" -pp "9091"
```

```
[root@host-142 /]# /opt/dks-wsks-3.1.24/install.sh -ws "nginx" -p "443" -cs "docker" -pp "9091"

~
~ Установка контейнера WEB-Сервер-КС 3.1.24
~ http сервер: nginx
~ порт : 443
~ Контейнеризация : docker
~

[ OK ] Проверка доступности порта 443 ПО "WEB-Сервер-КС".
[ OK ] Проверка доступности порта 9091 ПО "prometheus".
[ OK ] Проверка работы сервиса ПО "Docker" (docker.service).
[ OK ] Загрузка эталонного локального образа wsks-3.1.24 из архива ksdeb11_wsks-3.1.24_image.tar.
[ OK ] Построение конечного образа wsks-3.1.24_443 на базе эталонного.
Структура каталогов /opt/wsks-3.1.24_443 для запуска контейнера wsks-3.1.24_443 не существует, создаем ...
[ OK ] Создание структуры каталогов для запуска контейнера wsks-3.1.24_443.
[ OK ] Создание контейнера wsks-3.1.24_443 и запуск в docker (порт 443).
[ OK ] Создание пользователя ks-www-data.
Группа ks-www-data существует.
Пользователь ks-www-data существует.
[ OK ] Остановка контейнера wsks-3.1.24_443.
[ OK ] Настройка ротации логов ПО "WEB-Сервер-КС".
[ OK ] Запуск сервиса wsks-3.1.24_443.service.
[ Инфо ] Остановка сервиса wsks-3.1.24_443.service.
Отсутствуют установленные приложения.
Для развертывания веб приложений используйте скрипт ws_addapp.sh.
См. инструкцию по развертыванию /opt/wsks-3.1.24_443/ctl/readme.txt
```

9. Разверните .NET Core приложение "Бюджет-NEXT" для работы с СУБД PostgreSQL с именем сервера "127.0.0.1:5432" и базы данных "test", Обращение к сайту обеспечить через виртуальный каталог "bnext".

```
wget -O /opt/wsks-3.1.24_443/ctl/tar_files/webnext.tar.gz "https://keysystems.ru/files/web/INSTALL/BUDGET_NEXT/23.1/webnext-231.tar.gz" --no-check-certificate
```

```
bash /opt/wsks-3.1.24_443/ctl/ws_addapp.sh -t "wBKS" -a "webnext.tar.gz" -vc "bnext" -st "3" -sn "127.0.0.1:5432" -dn "test"
```

```
[root@host-142 /]# wget -O /opt/wsks-3.1.24_443/ctl/tar_files/webnext.tar.gz "https://keysystems.ru/files/web/INSTALL/BUDGET_NEXT/23.1/webnext-231.tar.gz" --no-check-certificate
--2023-03-30 16:53:05-- https://keysystems.ru/files/web/INSTALL/BUDGET_NEXT/23.1/webnext-231.tar.gz
Распознаётся keysystems.ru (keysystems.ru)... 192.168.4.214
Подключение к keysystems.ru (keysystems.ru)[192.168.4.214]:443 ... соединение установлено.
HTTP-запрос отправлен. Ожидание ответа... 200 OK
Длина: 162822518 (155M) [application/x-gzip]
Сохранение в: </opt/wsks-3.1.24_443/ctl/tar_files/webnext.tar.gz>

/opt/wsks-3.1.24_443/ctl/tar_files/webnex 100%[=====] 155,28M 96,7MB/s за 1,6s
2023-03-30 16:53:06 (96,7 MB/s) - </opt/wsks-3.1.24_443/ctl/tar_files/webnext.tar.gz> сохрaнен [162822518/162822518]

[root@host-142 /]# bash /opt/wsks-3.1.24_443/ctl/ws_addapp.sh -t "wBKS" -a "webnext.tar.gz" -vc "bnext" -st "3" -sn "127.0.0.1:5432" -dn "test"

~
~ Установка прикладного приложения WEB-Сервер-КС
~

[ OK ] Проверка доступности порта 54430 ПО "Бюджет-NEXT".
[ OK ] Проверка виртуального каталога "bnext".
[ OK ] Создание каталога приложения /opt/wsks-3.1.24_443/www/html/wsks_54430.
[ OK ] Создание файла запуска приложения /opt/wsks-3.1.24_443/ctl/app/54430_app.sh.
Приложение будет доступно по адресу - https://10.38.46.142:443/bnext.
[ OK ] Создание файла конфигурации для приложения /opt/wsks-3.1.24_443/conf/ks/vhost.ks/54430_ks.conf.
[ OK ] Создание файла настроек для приложения /opt/wsks-3.1.24_443/www/html/wsks_54430/appsettings.Production.json.
[ OK ] Запуск сервиса wsks-3.1.24_443.service.

[root@host-142 /]#
```

... Приложение будет доступно по адресу - https://[хост]:443/bnext....

10. Проверьте работоспособность установленного приложения, открыв адрес в браузере.

