**Установка:**

Версия веб-клиент может измениться, актуальную версию смотреть здесь - <https://keysystems.ru/files/smeta/install/svod-smart/Linux/latest/>

1) Скачиваем необходимые файлы:  
# wget <https://keysystems.ru/files/smeta/install/svod-smart/Linux/dks-docker-ce-19.03.4.tar.gz>  (если до этого не был скачан)  
# wget <https://keysystems.ru/files/smeta/install/svod-smart/Linux/dks-wsks-3.1.15.tar.gz>  (если до этого не был скачан)  
# wget <https://keysystems.ru/files/smeta/install/svod-smart/Linux/latest/Keysystems.Svod.Web_netcore_20.2.0.35342.tar.gz>

2) Распаковываем в папку /opt:  
# tar -zxf dks-docker-ce-19.03.4.tar.gz -C /opt  
# tar -zxf dks-wsks-3.1.15.tar.gz -C /opt

3) Устанавливаем dks-docker-ce-19.03.4:  
# bash /opt/dks-docker-ce-19.03.4/install.sh

4) Создание экземпляра контейнера:  
# bash /opt/dks-wsks-3.1.15/install.sh -id "<**номер\_контейнера**>" (если контейнер с данным номером уже создан: указываем другой)  
- <**номер\_контейнера**> - Диапазон идентификаторов варьируется от 1 до 9.

5) Перемещаем архивы в tar\_files для дальнейшего развертывания приложения:  
# mv Keysystems.Svod.Web\_netcore\_20.2.0.35342.tar.gz /opt/wsks-3.1.15\_00<**номер\_контейнера**>/ctl/tar\_files/

6) Развертывание .Net Core приложения:  
# bash /opt/wsks-3.1.15\_00<**номер\_контейнера**>/ctl/ws\_addapp.sh -t "wSVOD:Keysystems.Svod.Web.Core.dll" -a "Keysystems.Svod.Web\_netcore\_20.2.0.35342.tar.gz" -p "<**порт**>" -st **"<1>"** -sn "<**2**>" -dn "<**3**>"

- <**порт**> - значение порта, по которому происходило развертывание .Net Core приложения диапазон портов от 5443 до 5493 включительно.

- <**1**> - тип источника СУБД(для PostgreSQL, это "3")

- <**2**> - имя источника СУБД

- <**3**> - база данных источника СУБД

7) ПРИ НЕОБХОДИМОСТИ ДОПОЛНИТЕЛЬНО выдаем доступ к файлам (права должны выдаваться при развертывании приложения):  
# chmod -R 755 /opt/wsks-3.1.15\_00<**номер\_контейнера**>/www/html/wsks\_<**порт**>  
# chmod -R 777 /opt/wsks-3.1.15\_00<**номер\_контейнера**>/www/html/wsks\_<**порт**>/temp  
И назначаем владельца:  
# chown -R www-data:www-data /opt/wsks-3.1.15\_00<**номер\_контейнера**>/www/html/wsks\_<**порт**>

8) Перезапускаем экземпляр контейнера:  
# systemctl restart wsks-3.1.15\_00<**номер\_контейнера**>.service

**Обновление существующего развернутого .NET Core приложения:**

Где брать архив обновления:  
- <https://keysystems.ru/files/smeta/install/svod-smart/Linux/latest/> (качаем сразу на Linux)

**NET 3.1.8:**  
- размещаем архив обновления .tar.gz по пути /opt/wsks-3.1.8\_00<**номер\_контейнера**>/ctl/tar\_files  
- для обновления развернутого по **5443** порту приложения на базе архива /opt/wsks-3.1.8\_00<**номер\_контейнера**>/ctl/tar\_files/Keysystems.Svod.Web\_netcore\_20.2.0.35342.tar.gz выполняем команду:  
# bash /opt/wsks-3.1.8\_00<**номер\_контейнера**>/ctl/ws\_updapp.sh "wSVOD" "Keysystems.Svod.Web\_netcore\_20.2.0.35342.tar.gz" "**5443**"

**NET 3.1.15:**  
- размещаем архив обновления .tar.gz по пути /opt/wsks-3.1.15\_00<**номер\_контейнера**>/ctl/tar\_files  
- для обновления развернутого по **5443** порту приложения на базе архива /opt/wsks-3.1.15\_00<**номер\_контейнера**>/ctl/tar\_files/Keysystems.Svod.Web\_netcore\_20.2.0.35342.tar.gz выполняем команду:  
# bash /opt/wsks-3.1.15\_00<**номер\_контейнера**>/ctl/ws\_updapp.sh -t "wSVOD" -a "Keysystems.Svod.Web\_netcore\_20.2.0.35342.tar.gz" -p "**5443**"

**ВАЖНО!!!** При обновлении файлы настроек и конфигураций appsettings.json и appsettings.Production.json не перезаписываются.  
**Необходимые файлы предварительно сохранить отдельно**, т.к. иначе перезапишутся.

**UPDATE!!!** Если в файл **/opt/wsks-3.1.15\_00<номер\_контейнера>/ctl/ws\_updapp.sh** добавлена подобная запись, то **storage.config предварительно сохранять не нужно**:

# копируем во временный каталог файлы настроек  
cp ${target\_app\_catalog}/appsettings.json ${target\_temp\_catalog}  
cp ${target\_app\_catalog}/appsettings.Production.json ${target\_temp\_catalog}  
cp ${target\_app\_catalog}/wskstype ${target\_temp\_catalog}  
**cp ${target\_app\_catalog}/storage.config ${target\_temp\_catalog}**  
echo -e "${sOK}Сохранение файлов настроек."

**ВАЖНО!!!** Архив обновления должен включать полный набор файлов приложений. Поставка изменений не допускается, т.к. обновление производится путем полного удаления содержимого рабочего каталога приложения и последующей записи файлов из архива обновления.

**Решение проблемы локализации (дата в корректном формате dd.MM.yyyy):**

1) Выводим список контейнеров:  
# docker ps  
2) Проваливаемся внутрь контейнера:  
# docker exec -it <**имя\_контейнера**> bash  
3) Внутри контейнера:  
# apt-get install locales  
# dpkg-reconfigure locales  
    Выбираем **ru\_RU.UTF8**  
# exit  
4) Перезапускаем контейнер:  
# systemctl restart <**имя\_контейнера**>.service

**Решение проблемы со временем(невозможно было войти в комплекс, был установлен часовой пояс America/NewYork):**

1) Выводим список временных зон:  
# timedatectl list-timezones  
2) Выбираем нужную:  
# timedatectl set-timezone Asia/Yekaterinburg  
3) Перезапускаем контейнер:  
# systemctl restart <**имя\_контейнера**>.service