Оглавление

[Установка 2](#_Toc81217584)

[Установка сопутствующих программ 2](#_Toc81217585)

[Работа с консолью 5](#_Toc81217586)

[Проверка наличия текстового редактора 5](#_Toc81217587)

[Создание нового пользователя с домашней папкой 5](#_Toc81217588)

[Установка среды выполнения 8](#_Toc81217589)

[Установка фаервола 9](#_Toc81217590)

[Установка веб-сервера nginx 10](#_Toc81217591)

[Красным выделены пути сертификатов, которых еще нет, поэтому перед ними стоят символы «#». 11](#_Toc81217592)

[Установка системного часового пояса 13](#_Toc81217593)

[Установка базы данных 13](#_Toc81217594)

[Создание сертификата 16](#_Toc81217595)

[Теперь создадим сертификат 16](#_Toc81217596)

[Создание сервиса 17](#_Toc81217597)

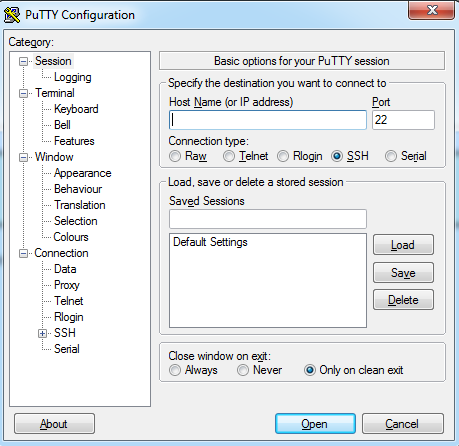
[Смена домена при развернутой системе 19](#_Toc81217598)

# Установка

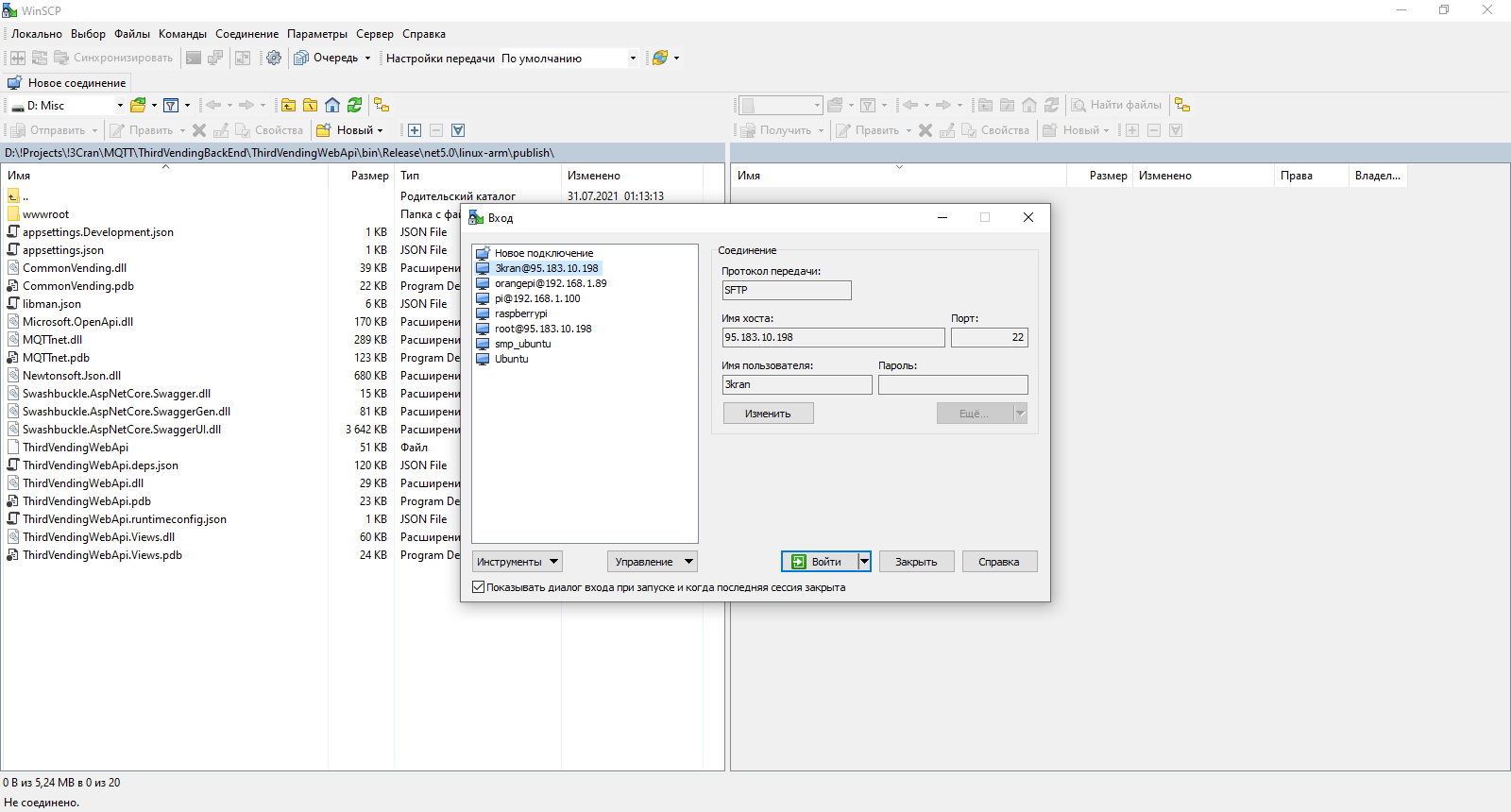
## Установка сопутствующих программ

Нам понадобятся:

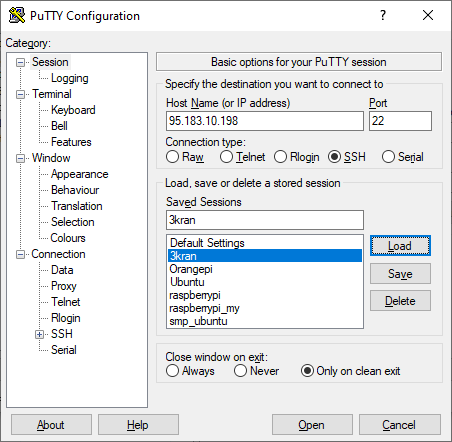
– Putty ([скачать](https://www.putty.org/)) – свободно распространяемый SSH клиент.



– WinSCP ([скачать](https://winscp.net/eng/download.php))– это графический клиент SFTP (SSH File Transfer Protocol) для Windows с открытым исходным кодом.



После скачивания и установки данных программ, запускаем Putty вводим IP адрес

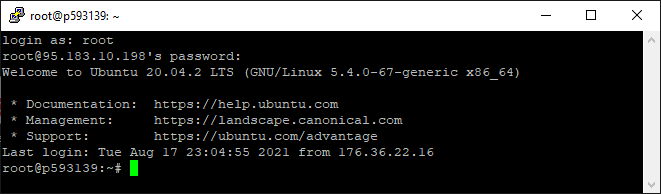


Нажимаем «Open»

Появится консоль



Вводи имя пользователя root, нажимаем на клавиатуре Enter, потом нужно ввести пароль



Чтобы вставить в консоль скопированный текст, нужно в консоли кликнуть правой клавишей мыши и текст вставится в положении курсора. Копирование из консоли производится при выделении текста мышью с зажатой левой клавишей мыши.

После подключения нужно обновить систему, поэтому по очереди ввести команды, после каждой дождаться выполнение, это может занять несколько минут

sudo apt-get update

sudo apt-get upgrade

sudo apt update

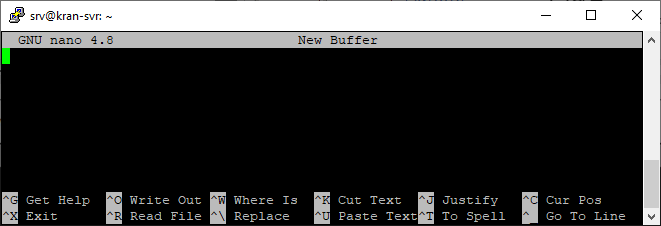
sudo apt upgrade

Это два разных пакетных менеджера.

# Работа с консолью

## Проверка наличия текстового редактора

Проверим наличие текстового редактора «nano», для этого введем в консоль nano, если откроется окно редактирования:



Тогда все нормально, если выдаст ошибку, тогда установим его, введя команду

sudo apt-get install nano

Он нам нужен для редактирования файлов конфигурации.

## Создание нового пользователя с домашней папкой

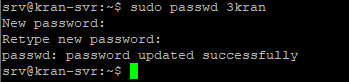
Для создания нового пользователя (3kran), нужно ввести команду

sudo useradd -m 3kran

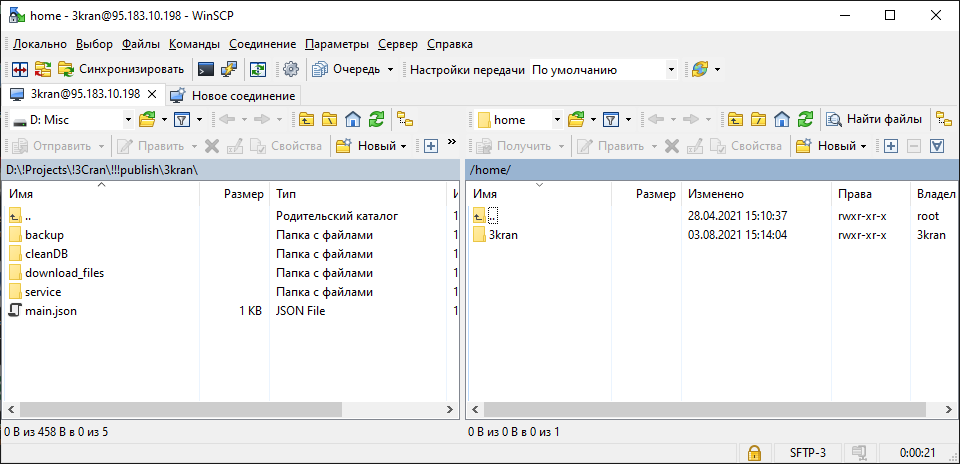
Теперь нужно создать пароль для этого пользователя, введя команду, где «3kran» пароль.

sudo passwd 3kran

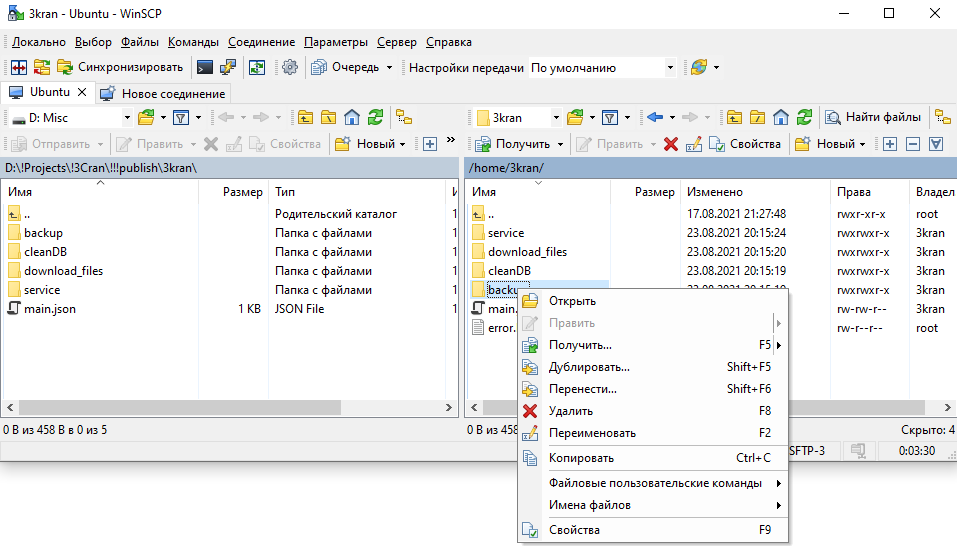
Введите пароль и повторите его



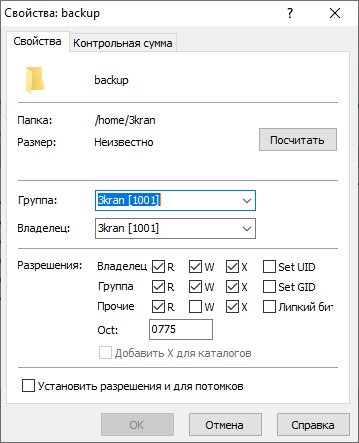
Теперь на сервере есть пользователь со своей домашней папкой «3kran», в которую нужно распаковать проект.



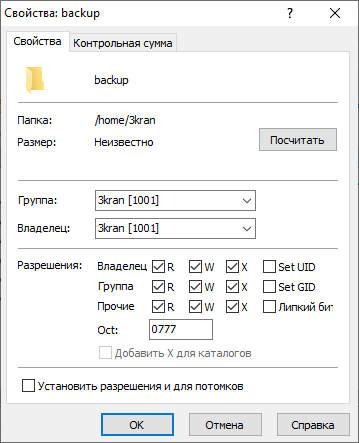
После копирования нужно проверить разрешения на папке «backup»



Нужно вызвать меню «Свойства»



И установить разрешения для всех



## Установка среды выполнения

Для установки версии с помощью APT потребуется выполнить несколько команд. Перед установкой .NET выполните приведенные ниже команды, чтобы добавить ключ подписывания пакета Майкрософт в список доверенных ключей и добавить репозиторий пакетов.

Откройте терминал и выполните следующие команды:

wget https://packages.microsoft.com/config/ubuntu/20.04/packages-microsoft-prod.deb -O packages-microsoft-prod.deb

sudo dpkg -i packages-microsoft-prod.deb

rm packages-microsoft-prod.deb

Среда выполнения ASP.NET Core позволяет запускать приложения, созданные с помощью версии .NET без поддержки среды выполнения. Приведенные ниже команды позволяют установить среду выполнения ASP.NET Core, которая больше всего совместима с .NET. В терминале выполните приведенные ниже команды.

sudo apt-get update; \

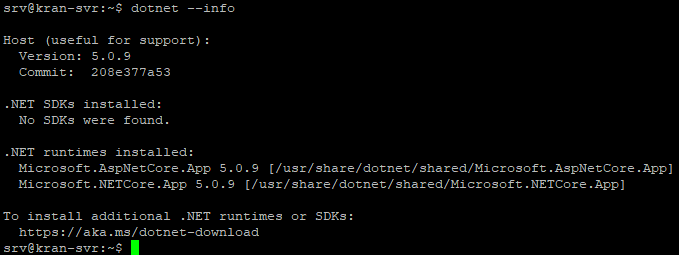
sudo apt-get install -y apt-transport-https && \

sudo apt-get update && \

sudo apt-get install -y aspnetcore-runtime-5.0

Проверим версию

dotnet --info



## Установка фаервола

Установим фаервол и откроем несколько портов

sudo apt-get install ufw

Разрешим 22 порт для работы по SSH, нужно обязательно, чтобы не отключился клиент SSH

sudo ufw allow 22/tcp

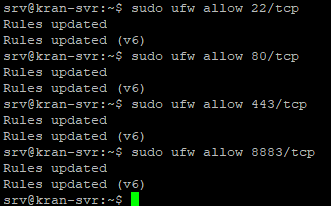
Разрешим 80 b 443 порты для работы веб браузера

sudo ufw allow 80/tcp

sudo ufw allow 443/tcp

Разрешим 8883 порт для работы MQTT

sudo ufw allow 8883/tcp



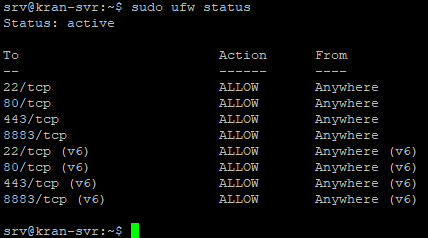
Включаем ufw фаервол

sudo ufw enable



Проверяем статус

sudo ufw status



## Установка веб-сервера nginx

sudo apt-get install nginx

Затем редактируем файл конфигурации

sudo nano /etc/nginx/sites-available/default

Все в файле удаляем и вставляем

server {

listen 80;

error\_log /home/3kran/error.log;

client\_max\_body\_size 1G;

location /send {

proxy\_pass http://localhost:5000;

proxy\_http\_version 1.1;

proxy\_set\_header Upgrade $http\_upgrade;

proxy\_set\_header Connection keep-alive;

proxy\_set\_header Host $host;

proxy\_cache\_bypass $http\_upgrade;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_set\_header X-Forwarded-Proto $scheme;

}

location /unixtimestamp {

proxy\_pass http://localhost:5000;

proxy\_http\_version 1.1;

proxy\_set\_header Upgrade $http\_upgrade;

proxy\_set\_header Connection keep-alive;

proxy\_set\_header Host $host;

proxy\_cache\_bypass $http\_upgrade;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_set\_header X-Forwarded-Proto $scheme;

}

location /{

return 301 https://$host$request\_uri;

}

}

server {

# listen 443 ssl;

listen 443;

error\_log /home/3kran/error.log;

client\_max\_body\_size 1G;

# ssl\_certificate /etc/letsencrypt/live/monitoring3voda.ru/fullchain.pem;

# ssl\_certificate\_key /etc/letsencrypt/live/monitoring3voda.ru/privkey.pem;

location / {

proxy\_pass http://localhost:5000;

proxy\_http\_version 1.1;

proxy\_set\_header Upgrade $http\_upgrade;

proxy\_set\_header Connection keep-alive;

proxy\_set\_header Host $host;

proxy\_cache\_bypass $http\_upgrade;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

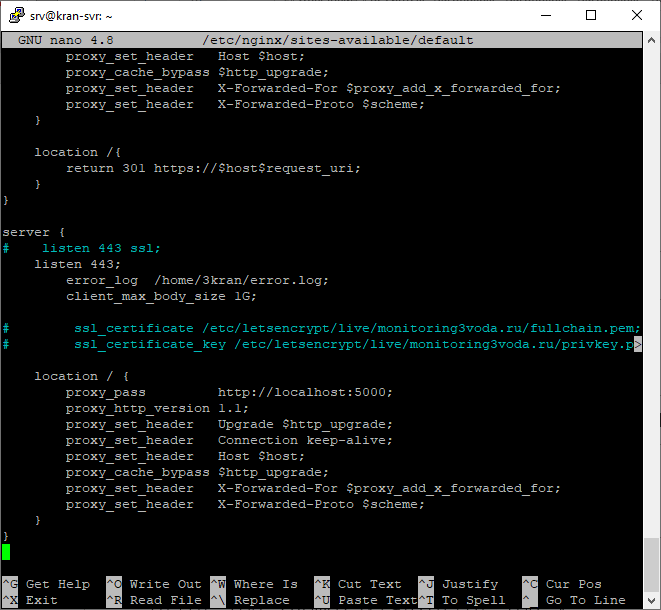
proxy\_set\_header X-Forwarded-Proto $scheme;

}

}

### Красным выделены пути сертификатов, которых еще нет, поэтому перед ними стоят символы «#».

После получения сертификата для своего домена, нужно еще раз отредактировав конфигурацию, удалив символы «#», и указав правильный каталог – доменное имя. А также изменить строчку «listen 443;» на «listen 443 ssl;»



Нажимаем Ctrl+X (Exit) и



Нажимаем Y



Потом Enter

Теперь проверим конфигурацию на ошибки

sudo nginx -t



Перезапускаем nginx

sudo service nginx restart

## Установка системного часового пояса

sudo timedatectl set-timezone Asia/Yekaterinburg

## Установка базы данных

Создайте конфигурацию файлового репозитория:

sudo sh -c 'echo "deb http://apt.postgresql.org/pub/repos/apt $(lsb\_release -cs)-pgdg main" > /etc/apt/sources.list.d/pgdg.list'

Импортируйте ключ подписи репозитория:

wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo apt-key add -

Обновите списки пакетов:

sudo apt-get update

Установите последнюю версию PostgreSQL.

Если вам нужна конкретная версия, используйте postgresql-13 или аналогичный вместо postgresql:

sudo apt-get -y install postgresql

Меняем часовой пояс для базы данных, для этого открываем файл настроек, папка 13 – это версия postgresql

sudo nano /etc/postgresql/13/main/postgresql.conf

Для поиска нажмите ctrl+w и введите «timezone»



‘Etc/UTC’ заменить на ‘Asia/Yekaterinburg’

Для поиска далее нажмите alt+w



Произвести такую же замену

Перезапускаем postgresql

sudo service postgresql restart

Заходим в консоль postgresql

sudo -u postgres psql

и вводим

create user kran with encrypted password 'qwerty';

где ‘qwerty’ пароль кототый повторяется в файле main.json

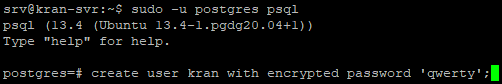
"ConnectionStrings": {

"Password": "qwerty"

}

Для смены настроек, нужно открыть файл main.json в редакторе:

sudo nano /home/3kran/main.json



Меняем роль для пользователя «kran»

ALTER USER kran SUPERUSER;

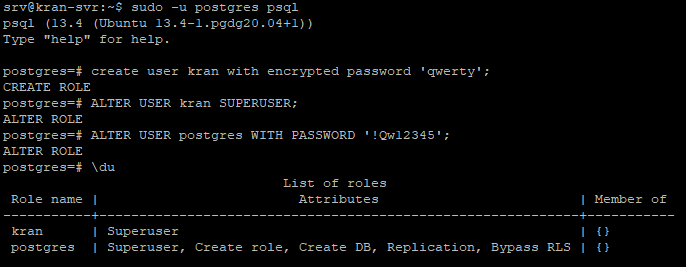
Меняем пароль для пользователя «postgres»

ALTER USER postgres WITH PASSWORD '!Qw12345';

Где ‘!Qw12345’ – ваш пароль

Проверяем список пользователей

\du



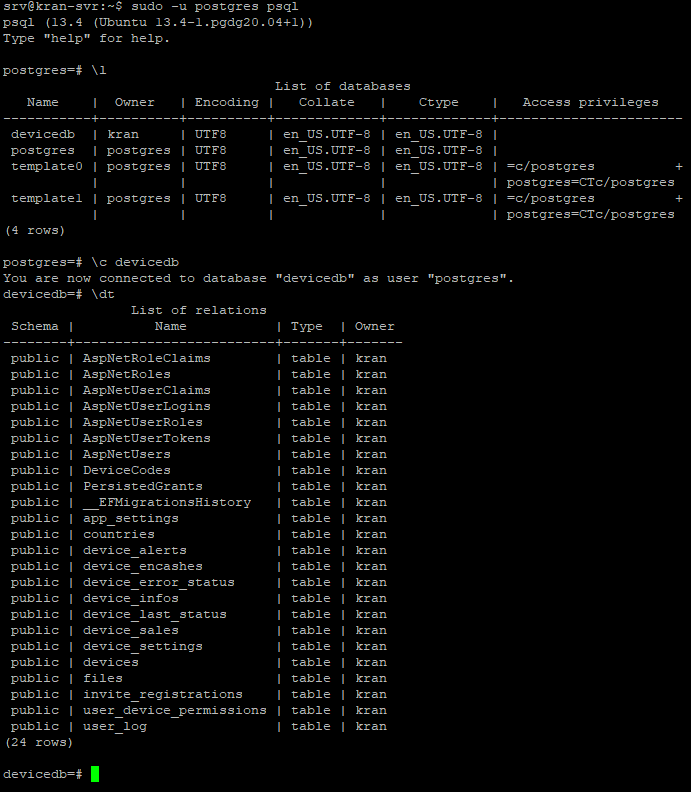
Нажимаем ctrl+z, чтобы выйти в основную консоль



Создаем чистую базу данных

sudo -u postgres createdb -w devicedb && sudo -u postgres pg\_restore -U postgres -w -F custom -d devicedb -C /home/3kran/cleanDB/deviceDB\_clean.back

После этого с помощью команд можно посмотреть на базу данных



## Создание сертификата

Обновите списки пакетов:

sudo apt update

Установите систему развёртки и управления пакетами

sudo apt install snapd

sudo snap install core; sudo snap refresh core

Установите клиент протокола ACME предназначенный для автоматического управления SSL-сертификатами от Let's Encrypt

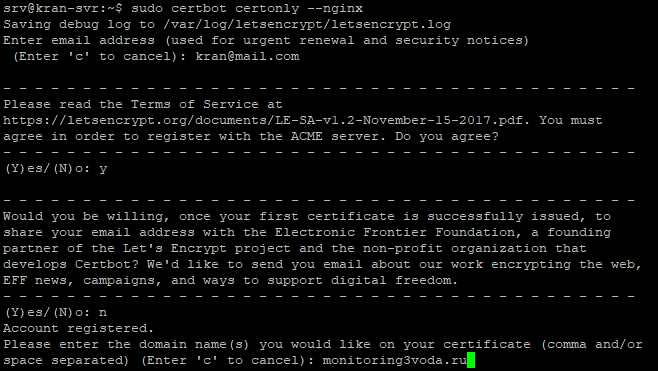
sudo snap install --classic certbot

Затем

sudo ln -s /snap/bin/certbot /usr/bin/certbot

### Теперь создадим сертификат

sudo certbot certonly --nginx

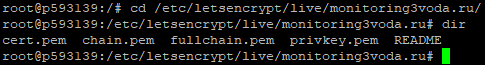


Введите свой или корпоративный имейл, согласитесь с условиями, отклоните следующий пункт, и в конце введите свое доменное имя, которое должно совпадать с именем и файла main.json, например

"MainHost": "monitoring3voda.ru"

Сертификат и ключ должны создаться по пути /etc/letsencrypt/live/{домен}, в данном случае

/etc/letsencrypt/live/monitoring3voda.ru



После создания сертификата нужно вернуться к настройкам nginx веб сервера и внести [правки](#_Красным_выделены_пути)

## Создание сервиса

Создайте файл «3kran-web.service»

sudo nano /etc/systemd/system/3kran-web.service

Скопируйте в него текст

[Unit]

Description=Monitoring 3kran system

DefaultDependencies=no

Wants=postgresql.service

After=postgresql.service

[Service]

RemainAfterExit=no

ExecStart=dotnet ThirdVendingWebApi.dll

WorkingDirectory=/home/3kran/service

SyslogIdentifier=3kran-web

Restart=always

RestartSec=30

KillSignal=SIGINT

[Install]

WantedBy=multi-user.target

Закройте с сохранением

Ведите команду

sudo systemctl daemon-reload

Потом активируйте сервис

sudo systemctl enable 3kran-web.service

Сделайте старт сервиса

sudo systemctl start 3kran-web.service

Проверьте его работу

sudo systemctl status 3kran-web.service

Результат

Aug 23 23:41:35 p593139.kvmvps systemd[1]: Started Monitoring 3kran system.

Aug 23 23:41:37 p593139.kvmvps 3kran-web[3209013]: info: Microsoft.Hosting.Lifetime[0]

Aug 23 23:41:37 p593139.kvmvps 3kran-web[3209013]: Now listening on: http://localhost:5000

Aug 23 23:41:37 p593139.kvmvps 3kran-web[3209013]: info: Microsoft.Hosting.Lifetime[0]

Aug 23 23:41:37 p593139.kvmvps 3kran-web[3209013]: Application started. Press Ctrl+C to shut down.

Aug 23 23:41:37 p593139.kvmvps 3kran-web[3209013]: info: Microsoft.Hosting.Lifetime[0]

Aug 23 23:41:37 p593139.kvmvps 3kran-web[3209013]: Hosting environment: Production

Aug 23 23:41:37 p593139.kvmvps 3kran-web[3209013]: info: Microsoft.Hosting.Lifetime[0]

Aug 23 23:41:37 p593139.kvmvps 3kran-web[3209013]: Content root path: /home/3kran/service

Проверьте еще раз настройки ngnix

sudo nginx -t



Перезапускаем nginx

sudo service nginx restart

Проверьте веб ресурс в браузере.

Все должно заработать!

# Смена домена при развернутой системе

При смене доменного имени нужно перевыпустить [сертификат](#_Теперь_создадим_сертификат) для него и поменять пути в настройках [nginx](#_Красным_выделены_пути), как же сменить доменное имя в настройках сервера main.json, например:

"MainHost": "3voda.money"

Для смены настроек, нужно открыть файл main.json в редакторе:

sudo nano /home/3kran/main.json

Сертификат и ключ должны создаться по пути /etc/letsencrypt/live/{домен}, в данном случае

/etc/letsencrypt/live/3voda.money

После правок нужно перезапустить сервис

sudo systemctl restart 3kran-web.service

Теперь проверим конфигурацию на ошибки

sudo nginx -t



Перезапускаем nginx

sudo service nginx restart