

Руководство по разворачиванию бота для Telegram

Оглавление

Введение	3
1. Регистрация бота Telegram	3
2. Установка бота Telegram	3
2.1. Windows.....	3
2.2. Linux	3
Установка среды выполнения .Net	4
Создание каталога для приложения.....	4
Автозапуск приложения.....	4
Установка Nginx.....	5
Конфигурирование Nginx	6
3. Настройка бота Telegram	7

Введение

Бот для Telegram реализован в виде .Net Core Web API приложения. Для работы под ОС Windows для простоты развертывания реализован в виде службы Windows.

1. Регистрация бота Telegram

Регистрация бота осуществляется при помощи встроенного в Telegram бота BotFather.

Для того, чтобы зарегистрировать нового бота, необходимо отправить в чат с BotFather команду `/newbot` и следовать инструкциям:

1. Придумать имя бота, которое будет отображаться в чатах и контактах.
2. Придумать username — имя должно быть уникальным и оканчиваться на «bot». Допускаются буквы латинского алфавита, цифры и символ подчеркивания. Общее количество символов не менее 5 и не более 32;
3. В ответ мы получим сообщение с токеном и ссылкой на бота. В дальнейшем эти данные понадобятся при настройке

2. Установка бота Telegram

2.1. Windows

Для работы модуля необходимо установить ASP.NET Core Runtime 3.1(<https://dotnet.microsoft.com/download/dotnet-core/3.1>)

Telegram бот может быть установлен на сервере отличном от сервера Pilot-ICE. Для установки бота Telegram необходимо запустить файл `Qwerta.Pilot.Telegram.Bot.Installer.msi`. В результате установке будет добавлена служба Windows - Pilot Telegram Bot.

2.2. Linux

Установка среды выполнения .Net

Перед установкой .NET выполните приведенные ниже команды, чтобы добавить ключ подписывания пакета Майкрософт в список доверенных ключей и добавить репозиторий пакетов.

```
sudo 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
```

Установка среды выполнения

```
sudo apt-get update
```

```
sudo apt-get install -y apt-transport-https
```

```
sudo apt-get update
```

```
sudo apt-get install -y aspnetcore-runtime-3.1
```

Создание каталога для приложения

```
sudo mkdir /var/www
```

```
sudo mkdir /var/www/telegrambot
```

Скопируйте содержимое архива Qwerta.Pilot.Telegram.Bot.zip в созданный каталог

Автозапуск приложения

Для автоматического запуска приложения необходимо создать юнит system.

Создаем и открываем для редактирования файл

```
sudo nano /etc/systemd/system/telegrambot.service
```

Содержимое файла:

```
[Unit]
Description=Pilot ICE Telegram Bot

[Service]
WorkingDirectory=/var/www/telegrambot
ExecStart=/usr/bin/dotnet /var/www/telegrambot/Qwerta.Pilot.Telegram.Bot.dll
Restart=always
RestartSec=10
SyslogIdentifier=telegrambot
User=root
Environment=ASPNETCORE_ENVIRONMENT=Production

[Install]
WantedBy=multi-user.target
```

После сохранения необходимо включить и запустить сервис при помощи следующих команд:

```
sudo systemctl enable telegrambot.service
```

```
sudo systemctl start telegrambot.service
```

Проверим статус сервиса:

```
sudo systemctl status telegrambot.service
```

Установка Nginx

В качестве обратного прокси сервере для обработки https запросов будем использовать Nginx

Для установки Nginx выполните команду:

```
sudo apt-get install -y nginx
```

Программа установки самостоятельно пропишет сценарии автозапуска systemd.

Так как Nginx устанавливается впервые, для запуска сервиса необходимо выполнить команду:

```
sudo service nginx start
```

Конфигурирование Nginx

В качестве адреса хоста в инструкции будет рассмотрен адрес example.ru. Вместо этого адреса вам необходимо указать свой адрес хоста.

Создайте файл конфигурации Nginx для вашего хоста:

```
sudo nano /etc/nginx/conf.d/example.ru.conf
```

Содержимое файла:

```
server {  
    listen    80;  
    server_name example.ru;  
    location / {  
        include conf.d/example.ru.inc;  
    }  
}
```

```
sudo nano /etc/nginx/conf.d/ example.ru.inc
```

Содержимое файла:

```
location / {  
    proxy_pass http://localhost:5000;  
    proxy_http_version 1.1;  
    proxy_set_header Connection keep-alive;  
    proxy_set_header Host $host;  
    proxy_set_header X-Forwarded-Host $http_host;  
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;  
    proxy_set_header X-Forwarded-Proto $scheme;  
    proxy_pass_header Sec-WebSocket-Extensions;  
}
```

3. Настройка бота Telegram

Настройка производится с помощью консольного приложения Qwerta.Pilot.Telegram.Bot.Configuration.exe расположенного в каталоге с установленным ботом Telegram. Необходимо запустить консоль и ввести запрашиваемые параметры:

- Token – токен полученный при регистрации бота
- Listen port - прослушиваемый порт. По умолчанию указать 8888
- Proxy address - адрес прокси-сервера (необязательный параметр)
- Proxy port - порт прокси сервера (необязательный параметр)
- Proxy user – имя пользователя для аутентификации на прокси-сервере (необязательный параметр)
- Proxy password -пароль для аутентификации на прокси-сервере (необязательный параметр)

После внесения параметров необходимо:

- для работы на ОС Windows – перезапустить службу бота (рис. 3.1)
- для работы на ОС Linux – перезапустить сервис бота

```
sudo service telegrambot restart
```

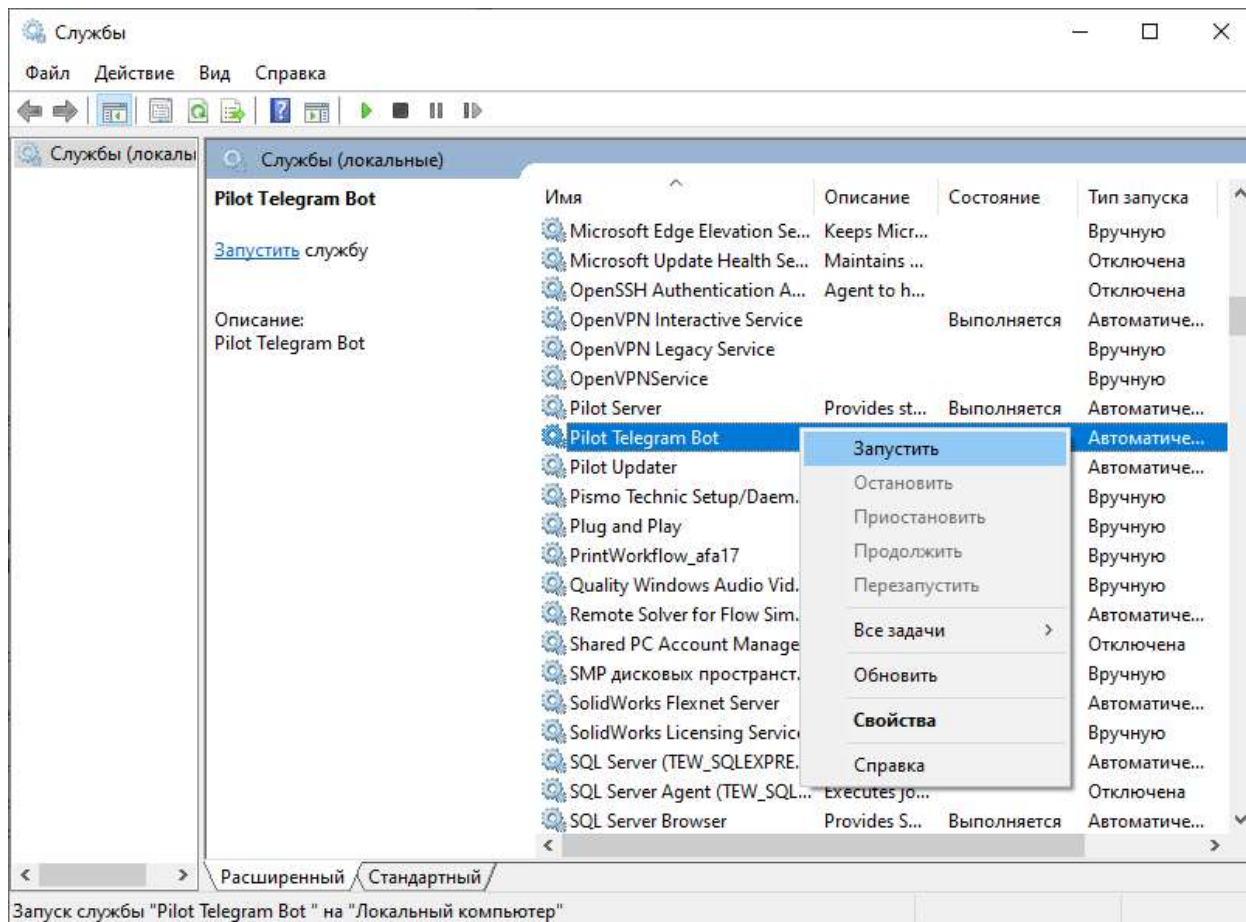


Рис 3.1 Запуск службы бота

Для проверки работоспособности бота введите в адресную строку браузера адрес сервера и прослушиваемый порт +/api/test (например <http://localhost:8888/api/test>) В результате должны увидеть название, версию приложения и имя пользователя бота (рис. 3.2).

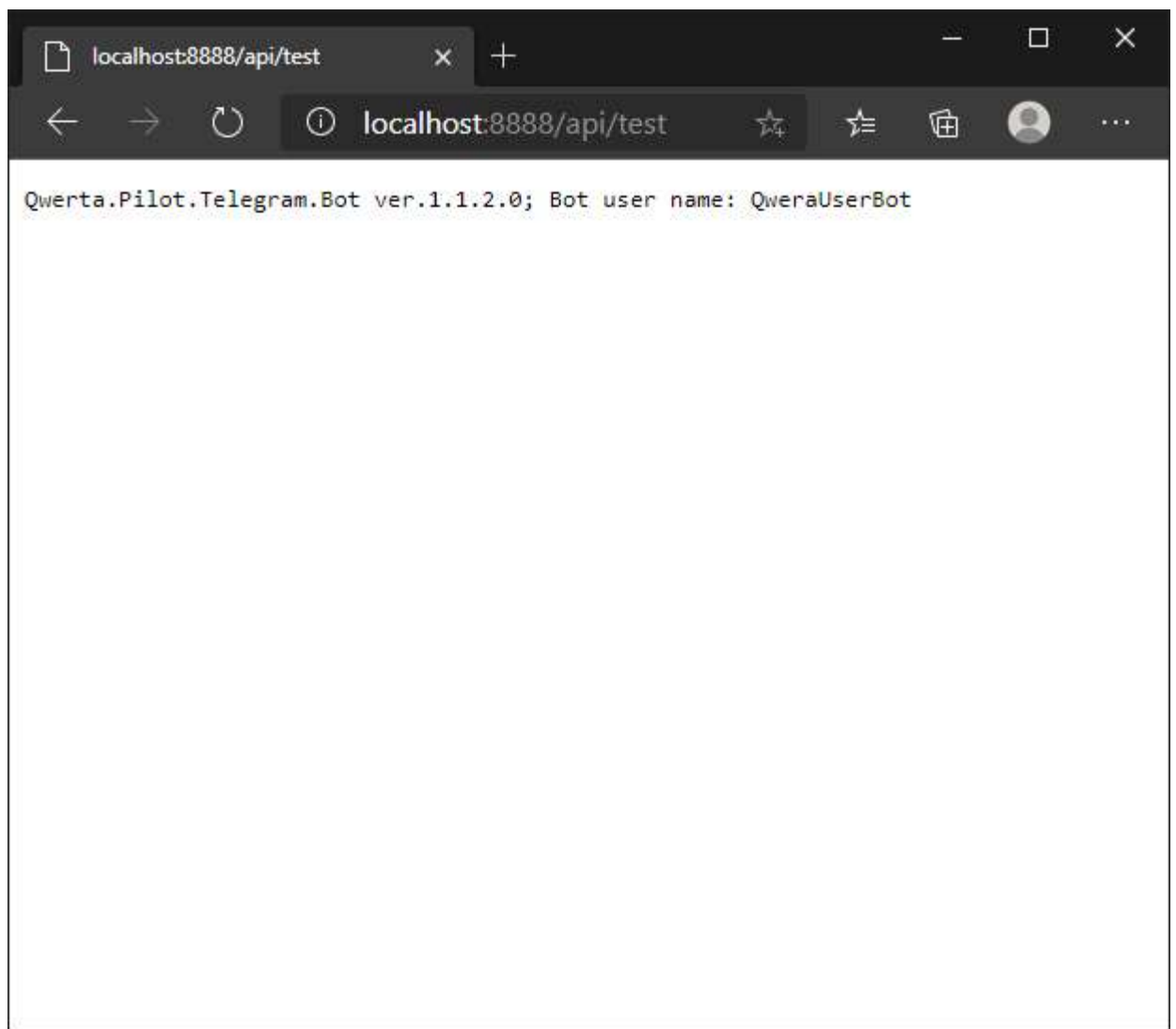


Рис 3.2. Проверка работоспособности бота