**Asp.net core от создания до запуска на nginx**

Для примера будем разворачивать приложение по-умолчанию – ASP.NET MVC Core с авторизации по пользователям приложения (ASP.NET Identity).

ОС для развертывания приложения: Ubuntu 18.04.1 LTS (Bionic Beaver).

БД для авторизации: MySQL 5.7.

Обратный прокси-сервер для пересылки запросов на веб-сервер Kestrel: nginx.

**Ubuntu. Установка mc**

После подключения по SSH, установим менеджер файлов mc, для этого:

**$ sudo add-apt-repository universe**

**$ then update**

**$ sudo apt update**

**$ sudo apt install mc**

Готово.

**Установка FTP-сервера**

Для того чтобы проводить развертывание приложения по ftp с помощью Visual Studio необходимо:

sudo apt install vsftpd

Настроить файл конфигурацииvsftpd.conf, открываем файл

sudo nano /etc/vsftpd.conf

и настраиваем:

anonymous\_enable = NO

local\_enable = YES

write\_enable = YES

local\_umask = 022

dirmessage\_enable = YES

xferlog\_enable = YES

xferlog\_std\_format=YES

connect\_from\_port\_20 = YES

listen=YES

pam\_service\_name=vsftpd

userlist\_enable = YES

userlist\_file=/etc/vsftpd.userlist

userlist\_deny=NO

chroot\_local\_user = YES

allow\_writeable\_chroot = YES

Далее рестартуем vsftpd:

**sudo systemctl restart vsftpd**

Настройка пользователя ftpuser:

**sudo useradd -m -c "ftp user" -s /bin/bash ftpuser**

**sudo passwd ftpuser**

**echo "ftp user" | sudo tee -a /etc/vsftpd.userlist**

**cat /etc/vsftpd.userlist**

**Установка dotnet**

**wget -q https://packages.microsoft.com/config/ubuntu/18.04/packages-microsoft-prod.deb**

**sudo dpkg -i packages-microsoft-prod.deb**

**sudo add-apt-repository universe**

**sudo apt-get install apt-transport-https**

**sudo apt-get update**

**sudo apt-get install dotnet-runtime-2.2**

**Установка Nginx**

**sudo apt-get install nginx**

**sudo service nginx start**

Проверка: http://<server\_IP\_address>/index.nginx-debian.html

Настройки /etc/nginx/sites-available/default:

server {

listen 80;

server\_name example.com \*.example.com;

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;

}

}

Запускаем приложение для проверки:

**dotnet <app\_assembly.dll>**

**Настройка демона автозапуска**

Пользователь www-data

**sudo nano /etc/systemd/system/kestrel-helloapp.service**

[Unit]

Description=Example .NET Web API App running on Ubuntu

[Service]

WorkingDirectory=/var/www/helloapp

ExecStart=/usr/bin/dotnet /var/www/helloapp/helloapp.dll

Restart=always

# Restart service after 10 seconds if the dotnet service crashes:

RestartSec=10

KillSignal=SIGINT

SyslogIdentifier=dotnet-example

User=www-data

Environment=ASPNETCORE\_ENVIRONMENT=Production

Environment=DOTNET\_PRINT\_TELEMETRY\_MESSAGE=false

[Install]

WantedBy=multi-user.target

**sudo systemctl enable kestrel-app.service**

**sudo systemctl start kestrel-app.service**

**sudo systemctl status kestrel-app.service**

**Установка mySQL**

sudo apt-get install mysql-server

Настройка безопасности:

Настройка внешнего доступа:

sudo nano /etc/mysql/my.cnf

bind-address = 0.0.0.0 (либо свой клиентский IP)

sudo service mysql restart

sudo mysql -p

*GRANT ALL PRIVILEGES ON \*.\* TO 'app'@'192.168.15.11' IDENTIFIED BY '123';*