

Szybki Kurs: nginx jako proxy_pass

Cel kursu

Celem kursu jest nauczenie Sebastiana pracy z nginx jako frontent przed aplikacja (javova, pythonowa, dowolna)

Wstęp

Najprostsza konfiguracja nginx służącego jako proxy pass wygląda tak:

```
server {
    listen 80;
    server_name twoja-domena.pl;

    location / {
        proxy_pass http://IP:PORT;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
    }
}
```

listen – mówi na jakim porcie słucha nginx (można podać też na jakim ip **WAŻNE!**)

server name – nazwa domenowa (używana kiedy masz wiele domen na jednym ip **WAŻNE!**)

location – najczęściej powiązane z proxy_pass mówi jakie adresy łapać i gdzie je przekierować (**NAJWAŻNIEJSZE**)

i tak / łapie wszystko i stronę główną / i każdą podstronę /admin, /css/ itp.)

Można to dowolnie mieszać

Reszta to ustawienia nagłówków

Najprostsza konfiguracja nginx służącego jako proxy pass z SSL wygląda tak:

```
server {
    listen 443 ssl;
    server_name twoja-domena.pl;

    # Ścieżki do Twoich certyfikatów
    ssl_certificate      /etc/certs/fullchain.pem;
    ssl_certificate_key  /etc/certs/privkey.pem;

    location / {
```

```

proxy_pass http://IP:PORT;
proxy_set_header Host $host;
proxy_set_header X-Real-IP $remote_addr;

# Dodatkowe nagłówki dla HTTPS
proxy_set_header X-Forwarded-Proto $scheme;
}
}

```

Jak widać różnica jest niewielka

2. Zadanie główne dla Sebastiana

- Utwórz 1 plik docker compose
- Uruchom w nim 2 kontenery z obrazami
 1. spidero/simple-app
 2. spidero/simple-api
- obrazy maja aplikacje działająca na porcie 8000
 1. Zmapuj te porty i wystaw aplikacje na swoim hoście, połącz się do nich przeglądarka (zrob screeny)
- Dodaj kolejny kontener z obrazem nginx i zmapuj pliku konfiguracyjnym w nim 2 location (pamiętaj żeby w dockerze używać nazw domenowych serwisu dockera a nie IP!!)
 1. location / > proxy_pass na kontener z obrazem simple-app
 2. location /api > proxy_pass na kontener z obrazem simple-api
 3. Usun porty wystawiające aplikacje simple-app i simple-api (dostęp do aplikacji ma być tylko przez nginx !!!)
 4. Wystaw nginx na jakiś porcie
 5. Połącz się do nginx przeglądarka i załącz screeny

3. Zadanie z *

Dodaj kolejny kontener ale z nginxem działającym po ssl

- pamiętaj że nginx wew mapuje się na 443 (najczęściej) (sekcja listen)
- pamiętaj że wchodząc na strone z sslem wpisywac <https://ip:port> bo inaczej przeglądarka sypnie ci błędem w stylu

400 Bad Request

The plain HTTP request was sent to HTTPS port

nginx/1.29.4

3. Zadanie super trudne **

Jesteś w stanie dodać nginx jako frontend dla naszej poprzedniej aplikacji z haproxy?