ТЕХНОЛОГИЧНО УЧИЛИЩЕ ЕЛЕКТРОННИ СИСТЕМИ

към ТЕХНИЧЕСКИ УНИВЕРСИТЕТ - СОФИЯ

КУРСОВ ПРОЕКТ

Предмет: Операционни системи

Тема: Създаване на работеща инстанция на WordPress

Ученик: *Калоян Георгиев*

Научен ръководител: *Кристиян Йочев*

СОФИЯ

2 0 2 5

1. Описание на подхода

1.1 Компоненти

1.1.1 Public Proxy (Nginx) – единствената услуга, достъпна от хост машината и външната мрежа.

1.1.2 Application (WordPress + PHP-FPM) – изпълнява логиката на приложението и не е директно достъпна отвън.

1.1.3 Database (MariaDB) – съхранява данните и е достъпна само от приложната среда.

1.2 Използвани технологии

1.2.1 Docker Compose

- Използван за създаване и управление на отделни контейнери и мрежи.
- Позволява дефиниране на изолирани мрежови сегменти за всеки компонент.

1.2.2 Nginx

- Обслужва публичния достъп (reverse proxy).
- Конфигуриран да препраща трафик към WordPress и да блокира директен достъп до базата данни.
- Избран пред Apache, поради възможността да се справя с по-висок трафик, скорост и проста конфигурация.

1.2.3 PHP-FPM

- Изпълнява PHP код на WordPress.
- Изолиран от публичния достъп.
- Подходящ за сървъри с общо предназначение, споделен хостинг и уеб сайтове с променлив трафик.

- Предимства:
    - Ефективно използване на паметта.
    - Добра производителност при променливо натоварване.
- **Недостатъци:**
    - Лек overhead при мащабиране на процесите.

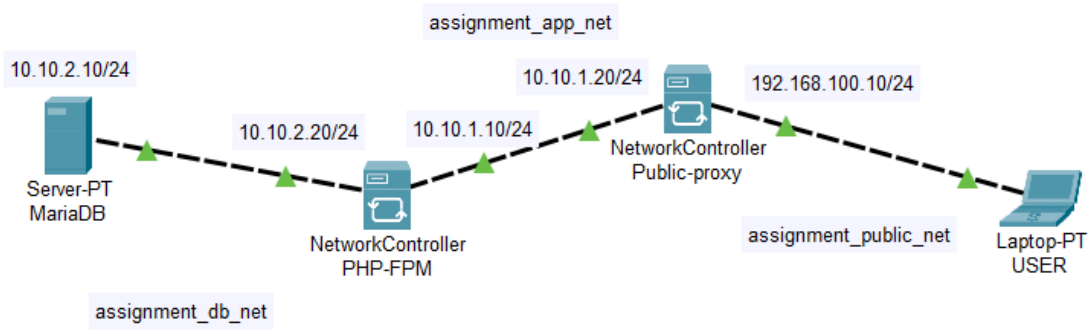
1.2.4 MariaDB

- База данни за WordPress.
- Изолация: не е достъпна директно от публичния контейнер или хост машината.
- Избрана пред MySQL заради по-чести ъпдейти за сигурност и по-добро справяне с едновременни заявки.

---

2. Мрежова топология

- **Диаграма на мрежата**



- **assignment_public_net** – Public Proxy (192.168.100.10/24)
    - Единствен достъп за хост машината и външни потребители


```
        "com.docker.compose.network": "public_net",
        "com.docker.compose.project": "assignment",
        "com.docker.compose.version": "5.0.0"
    },
    "Containers": {
        "f41c3a9964c9f14d817cf67d52ccb8298851f2c6f9eb4750c8d24b93c1c6eece": {
            "Name": "public-proxy",
            "EndpointID": "3342e66414e3539d84721d3e80c12fcc59d121f53cfe4f7ae6891c76197debda",
            "MacAddress": "2a:2a:e6:14:64:37",
            "IPv4Address": "192.168.100.10/24",
            "IPv6Address": ""
        }
```

- **assignment_app_net** – WordPress (10.10.1.0/24) + Public Proxy
    - WordPress комуникира само с public-proxy през app_net.

- **assignment_db_net** – MariaDB (10.10.2.0/24)
    - Достъпна само от WordPress.

---

3. Преглед на конфигурацията

3.1 Как е създадена всяка среда

Всяка среда е изградена като отделен Docker container с ясно дефинирана роля и изолация. Използван е Docker Compose за централизирано описание и управление на услугите.

3.1.1 Public Proxy (Nginx)

Публично-достъпната среда е реализирана чрез Nginx container, който приема входящия HTTP трафик от хост машината и външни потребители. Nginx работи като reverse proxy и препраща заявките към WordPress приложението. Само този container има отворени портове към хоста.

3.1.2 WordPress (PHP-FPM)

WordPress е разположен в отделен application container, който изпълнява PHP кода чрез PHP-FPM. Тази среда няма директен достъп от хост машината и комуникира единствено с public proxy и базата данни през вътрешни Docker мрежи.

3.1.3 MariaDB

Базата данни е стартирана в самостоятелен database container. Тя не е публично достъпна и приема връзки само от WordPress контейнера. Данните се съхраняват в отделен volume за персистентност при рестартиране на услугите.

3.2 Стартиране на услугите

Всички услуги се стартират и управляват чрез Docker Compose, който се стартира автоматично чрез systemd услуга при стартиране на операционната система.

Контейнерите са конфигурирани с restart policy, което гарантира автоматично стартиране при рестарт на Docker или на хост системата.

```
kaloyan@kaloyan-Virtual-Machine:~/os-coursework/assignment$ docker compose config | grep restart
    restart: unless-stopped
    restart: unless-stopped
    restart: unless-stopped
```

```
kaloyan@kaloyan-Virtual-Machine:~/os-coursework/assignment$ systemctl is-enabled docker
enabled
```

3.3 Конфигурация на потребители

За базата данни е създаден отделен потребител, използван единствено от WordPress, с права само върху съответната база данни. Не са предоставени административни или излишни привилегии.

Достъпът до базата данни е ограничен мрежово и е възможен само от контейнера на WordPress. Самата база данни не е публично достъпна.

Параметрите за връзка с базата данни са зададени в конфигурацията на WordPress, което осигурява контролирана и сигурна комуникация между услугите.

---

4. Мерки за сигурност

4.1 Изолация на средите

Средите са изолирани, като са поставени в различни контейнери и отделни мрежи.
assignment_public_net – публична мрежа, достъпна от хост машината и външния свят. В нея е разположен Public Proxy, който служи като входна точка за HTTP и SSH трафик. Нито WordPress, нито базата данни имат директен достъп до тази мрежа.

assignment_app_net – вътрешна мрежа за WordPress и Public Proxy. WordPress комуникира само с Public Proxy през тази мрежа, което предотвратява директен достъп до приложението от външни потребители.

assignment_db_net – изолирана мрежа за MariaDB, достъпна само от WordPress контейнера. Това гарантира, че базата данни не може да бъде достигната директно от хоста или други контейнери извън тази мрежа.

4.2 Правила на firewall

Firewall-ът позволява трафик само на портове 80 и 2222 и блокира останалия.

```
kaloyan@kaloyan-Virtual-Machine:~/os-coursework/assignment$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), deny (routed)
New profiles: skip

To                         Action      From
--                         ------      ----
80/tcp                     ALLOW IN    Anywhere
2222/tcp                   ALLOW IN    Anywhere
80/tcp (v6)                ALLOW IN    Anywhere (v6)
2222/tcp (v6)              ALLOW IN    Anywhere (v6)
```

4.3 Ограничения на достъпа

- Ограничена свързаност между средите

```
kaloyan@kaloyan-Virtual-Machine:~/os-coursework/assignment$ curl -I --connect-timeout 3 http://localhos
t:9000
curl: (7) Failed to connect to localhost port 9000 after 0 ms: Connection refused
kaloyan@kaloyan-Virtual-Machine:~/os-coursework/assignment$ curl --connect-timeout 3 http://localhost:3
306
curl: (7) Failed to connect to localhost port 3306 after 0 ms: Connection refused
kaloyan@kaloyan-Virtual-Machine:~/os-coursework/assignment$ curl -I http://localhost
HTTP/1.1 200 OK
Server: nginx/1.28.0
Date: Sun, 14 Dec 2025 19:13:30 GMT
Content-Type: text/html; charset=UTF-8
Connection: keep-alive
X-Powered-By: PHP/8.2.29
Link: <http://localhost/index.php?rest_route=/>; rel="https://api.w.org/"

kaloyan@kaloyan-Virtual-Machine:~/os-coursework/assignment$
```

- Ограничения на файловата система (минимални права дадени)

```
-rw-r--r--  1 appuser appuser  2323 Jun 14  2023 wp-comments-post.php
-rw-r--r--  1 appuser appuser  3339 Aug 12 14:47 wp-config-sample.php
-rw-r--r--  1 appuser appuser  3620 Dec 13 22:49 wp-config.php
drwxr-xr-x  4 appuser appuser  4096 Dec 15 09:50 wp-content
-rw-r--r--  1 appuser appuser  5617 Aug  2  2024 wp-cron.php
drwxr-xr-x 31 appuser appuser 16384 Dec  2 18:35 wp-includes
-rw-r--r--  1 appuser appuser  2493 Apr 30  2025 wp-links-opml.php
-rw-r--r--  1 appuser appuser  3937 Mar 11  2024 wp-load.php
-rw-r--r--  1 appuser appuser 51437 Oct 29 10:37 wp-login.php
-rw-r--r--  1 appuser appuser  8727 Apr  2  2025 wp-mail.php
-rw-r--r--  1 appuser appuser 31055 Nov  7 12:42 wp-settings.php
-rw-r--r--  1 appuser appuser 34516 Mar 10  2025 wp-signup.php
-rw-r--r--  1 appuser appuser  5214 Aug 19 12:30 wp-trackback.php
-rw-r--r--  1 appuser appuser  3205 Nov  8  2024 xmlrpc.php
kaloyan@kaloyan-Virtual-Machine:~/os-coursework/assignment$ docker exec -it wordpress-db ls -l /var/www/
html
ls: cannot access '/var/www/html': No such file or directory
kaloyan@kaloyan-Virtual-Machine:~/os-coursework/assignment$ docker exec -it wordpress-db ls -l /var/lib/
mysql
total 139592
-rw-rw---- 1 1000 1000  17055744 Dec 15 10:11 aria_log.00000001
-rw-rw---- 1 1000 1000        52 Dec 15 10:11 aria_log_control
-rw-rw---- 1 1000 1000         2 Dec 14 09:43 c589b1a03b45.pid
-rw-rw---- 1 1000 1000         9 Dec 15 15:23 ddl_recovery-backup.log
-rw-rw---- 1 1000 1000         9 Dec 15 15:23 ddl_recovery.log
-rw-rw---- 1 1000 1000         2 Dec 14 10:59 df978428719f.pid
-rw-rw---- 1 1000 1000         2 Dec 15 15:23 ff4930c9208c.pid
-rw-rw---- 1 1000 1000      1408 Dec 15 10:11 ib_buffer_pool
-rw-rw---- 1 1000 1000 100663296 Dec 15 15:24 ib_logfile0
-rw-rw---- 1 1000 1000  12582912 Dec 15 10:11 ibdata1
-rw-rw---- 1 1000 1000  12582912 Dec 15 15:23 ibtmp1
-rw-rw---- 1 1000 1000         0 Dec 13 22:34 multi-master.info
drwx------ 2 1000 1000      4096 Dec 13 22:34 mysql
-rw-r--r-- 1 1000 1000        16 Dec 13 22:34 mysql_upgrade_info
drwx------ 2 1000 1000      4096 Dec 13 22:34 performance_schema
drwx------ 2 1000 1000     12288 Dec 13 22:34 sys
drwx------ 2 1000 1000      4096 Dec 13 22:52 wordpress
kaloyan@kaloyan-Virtual-Machine:~/os-coursework/assignment$
```

- Изпълнение на услугите като непривилигериовани потребители (non-root execution)

```
kaloyan@kaloyan-Virtual-Machine:~/os-coursework/assignment$ docker exec -it wordpress-app id
docker exec -it wordpress-db id
uid=1000(appuser) gid=1000(appuser) groups=1000(appuser)
uid=1000 gid=1000 groups=1000
kaloyan@kaloyan-Virtual-Machine:~/os-coursework/assignment$
```

- Ограничения на използваните ресурси

```
  container_name: wordpress-db
  mem_limit: 512m
  cpus: 0.5
  volumes:
```

```
  image: nginx:stable
  container_name: public-proxy
  volumes:
    - ./public-proxy/nginx-public.conf:/etc/nginx/conf.d/default.conf:ro
    - ./volumes/wp_data:/var/www/html:ro
  mem_limit: 512m
  cpus: 0.5
```

```
    build: ./web
    container_name: wordpress-app
    volumes:
      - ./volumes/wp_data:/var/www/html
      - ./web/nginx-app.conf:/etc/nginx/conf.d/default.conf:ro
      - ./web/php-fpm-pool.conf:/usr/local/etc/php-fpm.d/www.conf:ro
    mem_limit: 512m
    cpus: 0.5
```

- Изолирани мрежови стекове

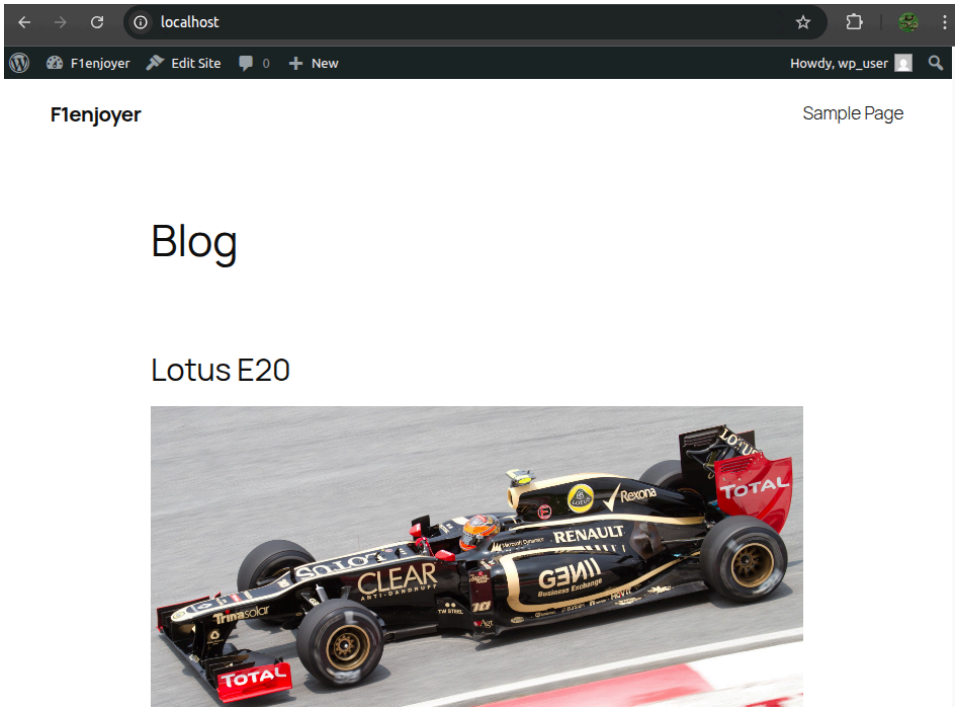
```
kaloyan@kaloyan-Virtual-Machine:~/os-coursework/assignment$ curl -I --connect-timeout 3 http://localhos
t:9000
curl: (7) Failed to connect to localhost port 9000 after 0 ms: Connection refused
kaloyan@kaloyan-Virtual-Machine:~/os-coursework/assignment$ curl --connect-timeout 3 http://localhost:3
306
curl: (7) Failed to connect to localhost port 3306 after 0 ms: Connection refused
kaloyan@kaloyan-Virtual-Machine:~/os-coursework/assignment$ curl -I http://localhost
HTTP/1.1 200 OK
Server: nginx/1.28.0
Date: Sun, 14 Dec 2025 19:13:30 GMT
Content-Type: text/html; charset=UTF-8
Connection: keep-alive
X-Powered-By: PHP/8.2.29
Link: <http://localhost/index.php?rest_route=/>; rel="https://api.w.org/"

kaloyan@kaloyan-Virtual-Machine:~/os-coursework/assignment$
```

---

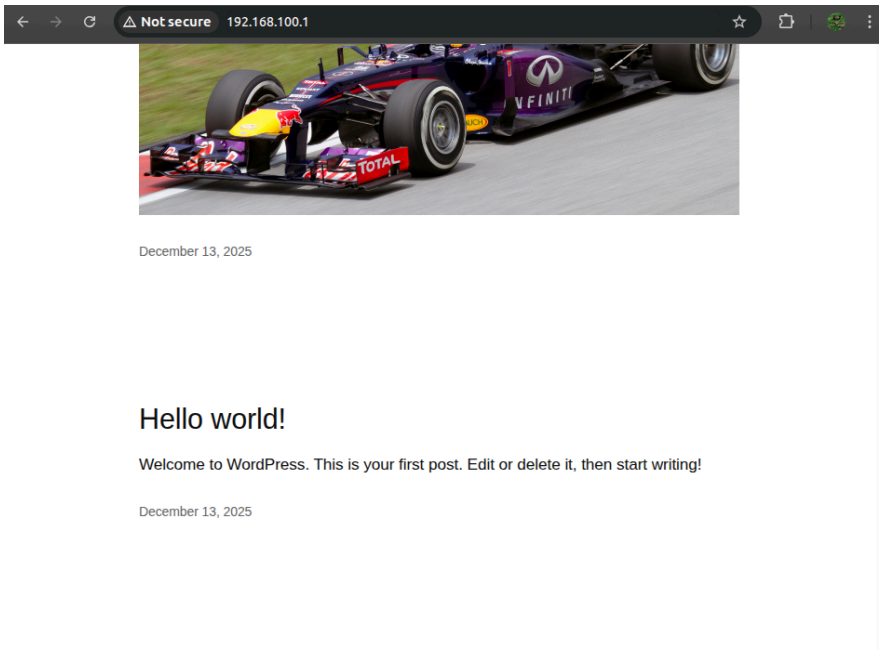
5. Тестове

5.1Показване на работоспособност

- Визуализация при http://localhost :



- Визуализация при http://192.168.100.1 :



5.2 Показване на забранени връзки

- Забранени са всякакви връзки от хостове към WordPress или MariaDB

```
kaloyan@kaloyan-Virtual-Machine:~/os-coursework/assignment$ curl -I --connect-timeout 3 http://localhos
t:9000
curl: (7) Failed to connect to localhost port 9000 after 0 ms: Connection refused
kaloyan@kaloyan-Virtual-Machine:~/os-coursework/assignment$ curl --connect-timeout 3 http://localhost:3
306
curl: (7) Failed to connect to localhost port 3306 after 0 ms: Connection refused
```

- Забранен е директен достъп от Public Proxy към MariaDB

```
kaloyan@kaloyan-Virtual-Machine:~/os-coursework/assignment$ docker exec -it public-proxy sh
# curl --connect-timeout 3 http://wordpress-db:3306
curl: (6) Could not resolve host: wordpress-db
#
```

- WordPress няма достъп до MariaDB извън assignment_db_net

```
kaloyan@kaloyan-Virtual-Machine:~/os-coursework/assignment$ docker exec -it wordpress-app sh
$ curl -I http://wordpress-db:3306
curl: (1) Received HTTP/0.9 when not allowed
```

Бележка: Изкуствен интелект е използван за търсене на източници и troubleshooting