



Лабораториска вежба бр. 4	Традиционална и прогресивна доставува на видеа		
Име и презиме	Индекс	Група	Датум

Користејќи го текстот од упатството, извршете ги потребните барања. Одговорите на барањата внесете ги во квизот за вежбата поставен на курсот.

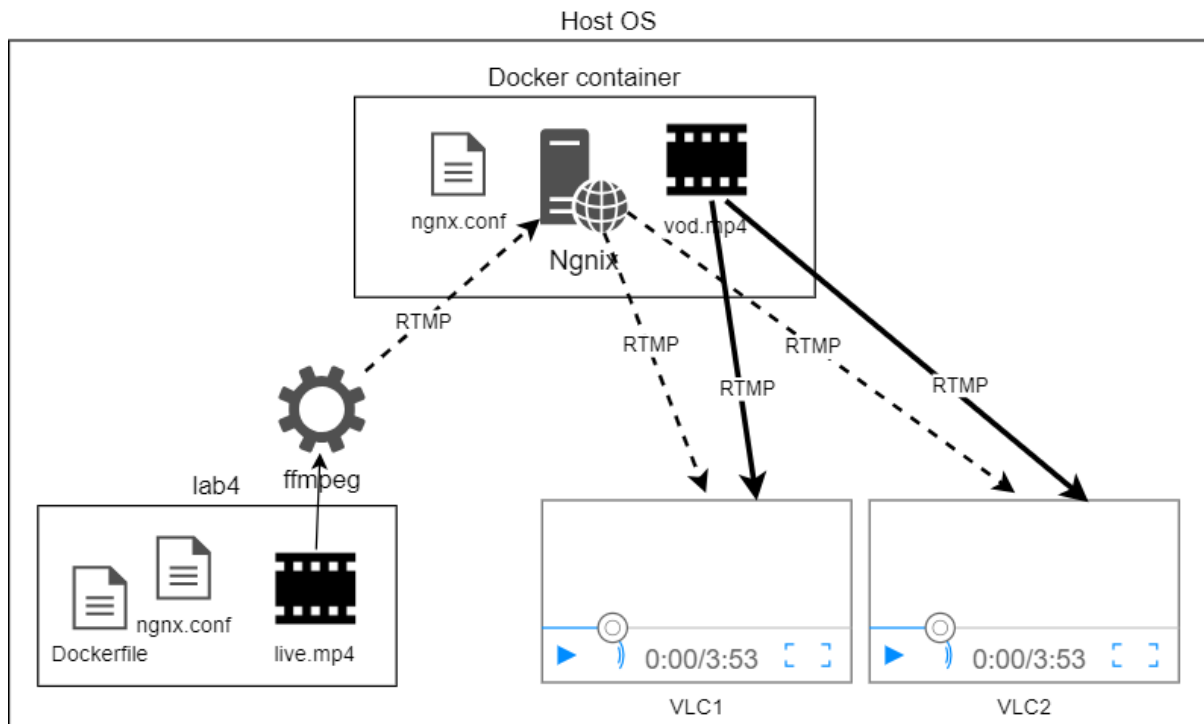
Лабораториската вежба не е задолжителна и се изработува самостојно!

Опис на архитектурата

За изработка на оваа вежба потребно е да се подигне сервер кој ќе прави традиционална и прогресивна доставува на видеа . Серверот nginx има модул за RTMP доставува, па затоа истиот ќе се користи во вежбата. За да се инсталира ваков сервер со минимален број на чекори и ресурси, ќе се креира виртуелна машина (контејнер) со помош на Docker од image на кој веќе ќе биде конфигурирана инсталација на Nginx со модул за доставување со RTMP. Овој контејнер ќе се извршува на вашиот оперативен систем. За да служи како сервер за достава на видеа во живо преку RTMP, потребно е серверот да се „нахрани“ со поток од настан во живо кој ќе го редистрибуира до сите корисници кои ќе го побараат. За таа цел, со користење на FFmpeg ќе се чита видео од локалниот диск на вашиот оперативен систем и ќе се доставува до nginx серверот, кој потоа истиот ќе го редистрибуира. За достава на VOD содржини, видеата се поставуваат како фајлови на серверот. За да ја демонстрирате исправноста на конфигурацијата, во вашиот оперативен систем ќе отворите неколку инстанци на VLC player и ќе го побарате потокот од серверот преку внесување на соодветна URL адреса.

Организацијата на сите компоненти е прикажана на Слика 1.

Во пракса, компонентата означена како Docker container ќе биде далечински сервер на дистрибутерот на видеа или Cloud сервис, а сите инстанци на VLC ќе бидат различните клиенти кои ќе ги користат видео сервисите. Ако станува збор за достава во живо, тогаш дистрибутерот на содржини ќе го доставува настанот во живо од дадена локација кон далечинскиот сервер за дистрибуција.



Слика 1 Архитектура на систем за достава на видеа

I Подготовка на работна околина

1. Потребно е да се инсталираат следниве алатки
 - a. Docker Desktop - <https://www.docker.com/get-started>
 - b. VLC - <https://www.videolan.org/vlc/>
2. Откако ќе ги инсталирате алатките, вклучете го Docker Desktop и во конзола (за Windows - PowerShell -> Run as administrator, за Linux - Terminal) внесете ја командата “docker” за да проверите дали успешно е активиран Docker. Доколку е успешно активиран, командата ќе ви испечати листа на команди што се користат со Docker.

II Креирање на Docker image и container

1. Симнете ја **.zip** датотеката во прилогот на лабораториската вежба. Поставете ја на позната локација (во понатамошниот текст за патеката ќе се користи **lokacija**). Направете extract на датотеката (содржи Dockerfile и nginx.conf).
2. Поставете ја работната патека во директориумот добиен после распакувањето (**lab4**) со командата
cd lokacija/lab4
 Овој директориум содржи **Dockerfile** датотека и **conf** директориум.

3. Додадете содржина на **nginx** конфигурацијата за nginx веб серверот так што ќе ја отворите датотеката **nginx.conf** (текстуален едитор по ваш избор) и ќе го додадете кодот во продолжение (конфигурација на модул за rtmp streaming и приказ на статистики) :

```
worker_processes 1;

events {
    worker_connections 1024;
}

rtmp {
    server {
        listen 1935;
        chunk_size 4000;
    }
}

# HTTP can be used for accessing RTMP stats
http {
    server {
        listen 8080;
        # This URL provides RTMP statistics in XML
        location /stats {
            rtmp_stat all;

            # Use this stylesheet to view XML as web page
            # in browser
            rtmp_stat_stylesheet /stat.xsl;
        }

        location /stat.xsl {
            # XML stylesheet to view RTMP stats.
            # Copy stat.xsl wherever you want
            # and put the full directory path here
            root /usr/local/nginx/html;
        }
        add_header Cache-Control no-cache;
    }
}
```

4. Во **server** секцијата од **rtmp** конфигурацијата на nginx.conf, внесете иницијална конфигурација за RTMP апликација (ќе слуша на порта 1935 да добие поток за редистрибуција и истиот ќе го редистрибуира) со име **rtmp-live-vash-index** со следната содржина (пр. rtmp-live-222111):

```
application rtmp-live-vash-index {

    # enable live streaming
    live on;

    # publish and play from any IP
    allow publish all;
    allow play all;
}
```



5. Зачувајте ја датотеката.
6. Одговорете го Dockerfile во директориумот lab4 (текстуален едитор). Под коментарот “# **Kopiranje na nginx config datoteka vo container**” додадете ја командата во продолжение. Со оваа команда датотеката nginx.conf ќе се копира на локација /etc/nginx/nginx.conf на docker image откако ќе го подигнете.

```
COPY nginx.conf /etc/nginx/nginx.conf
```

7. Зачувајте ги промените.
8. Направете build на Docker image-от од Dockerfile така што додека сте во директориумот lab4 ќе ја извршите следнава команда:

```
docker build -t lab4-<vashiot_indeks> .
```

- **-t** означува **tag**, односно името што го задавате на image-от, во овој случај, **lab4-<vashiot_indeks>**
 - Важно е да се запази **точката** на крајот на командата. Со тоа кажуваме на Docker да го креира image-от според Dockerfile-от во моменталниот директориум во кој што се наоѓаме
9. Откако ќе се креира image-от, со командата **docker image ls** може да добиете повеќе информации за него.
 10. За да креирате **container** од добиениот image, извршете ја командата **docker run -d -p 1935:1935 -p 8080:8080 lab4-<vashiot_indeks>**
 - со **-d** ќе добиеме контејнер во detached mode, односно ќе се извршува во позадина
 - со **-p** ги означуваме портите на кои го изложуваме контејнерот (1935 за rtmp, 8080 за http)
 - **lab4-<vashiot_indeks>** е името на image-от што го креиравте претходно.

11. Поставете исечок од екранот со резултатот на командата.
12. Проверете дали контејнерот е активен. Извршете ја командата:
docker ps
13. Поставете исечок од екранот со резултатот од командата. Доколку контејнерот не е активен, активирајте го со
docker start <container_id>



каде што `<container_id>` ќе го добиете со **docker ps** -а командата, што ги прикажува сите контејнери.

III Пренос на потоци во живо со RTMP

1. Со помош на **ffmpeg** испратете RTMP поток од .mp4 датотека од вашиот оперативен систем кон серверот (Слика 1). Симнете датотека **video.mp4** со траење од неколку минути (или искористете видео од претходните вежби). Зачувајте ја датотеката во директориумот `lokacija/lab4`.

2. Извршете ја следната команда:

```
ffmpeg -stream_loop -3 -re -i video.mp4 -f flv  
rtmp://localhost/rtmp-live-[vash_indeks]/live
```

Забелешка: параметарот `-re` поставува realtime вчитување на изворната датотека, `-stream_loop -3` го повторува видеото откако ќе заврши и тоа 3 пати, `-f flv` креира RTMP компатибилен поток на излезот.

3. Во прелистувач, отворете ја адресата <http://localhost:8080/stats> за да ги проверите статистиките за потокот и за да се уверите за исправноста на конфигурацијата.
4. Во алатката VLC (Media->Open network stream), внесете ја следната патека за преземање на rtmp потокот во живо:
rtmp://localhost:1935/rtmp-live-[vash-indeks]/live
5. Повторете го претходниот чекор за нова инстанца од VLC.
6. Обновете ја содржината на прелистувачот со статистики.
7. Направете исечок од екранот со двете инстанци на VLC и страната за статистики и потоа затворете ги прозорците на VLC.
8. Дадете кратка дискусија за двата по тока (од кој момент почнуваат, што се случува при пауза и повторно пуштање, враќање назад, одење напред)

IV Пренос на VoD потоци со RTMP

1. Во конфигурациската датотека **nginx.conf**, внесете го следниот конфигурациски блок во **server** секцијата на **rtmp** конфигурацијата:

```
# video on demand  
application rtmp-vod-vash-index {  
    play /usr/local/nginx/html/rtmp-vod;
```



```
}
```

- Во Dockerfile-от под коментарот # **RTMP Video-on-demand** додадете ја следната команда (видеото се копира од вашиот оперативен систем на локација во контејнерот од која веб серверот ги чита видеата)

```
COPY video.mp4 /usr/local/nginx/html/rtmp-vod/
```

- Запишете ги промените во Docker контејнерот така што ќе направите build на новиот image и повторно ќе го вклучите на контејнерот користејќи ги следните команди:

```
docker stop [container_id]  
docker build -t lab4-<vashiot_indeks> .  
docker run -d -p 1935:1935 -p 8080:8080 lab4-<vashiot_indeks>
```
- Во VLC отворете ја патеката **rtmp://localhost:1935/rtmp-vod-vash-index**
- Повторете го претходниот чекор за нова инстанца од VLC.
- Обновете ја содржината на прелистувачот со статистики.
- Направете исечок од екранот со двете инстанци на VLC и страната за статистики и потоа затворете ги прозорците на VLC.
- Дадете кратка дискусија за двата потока (од кој момент почнуваат, што се случува при пауза и повторно пуштање, враќање назад, одење напред)

V Пренос на прогресивни Video-on-demand потоци преку HTTP

- Во VLC отворете ја патеката (пристап до истото видео од претходниот чекор, но преку протоколот HTTP)

```
http://localhost:8080/vod/video.mp4
```

- Внесете ја истата патека во веб прелистувач.
- Направете исечок од екранот на VLC и прелистувачот.
- Дадете кратка дискусија за двата потока (од кој момент почнуваат, што се случува при пауза и повторно пуштање, враќање назад, одење напред). Што се случи кога патеката се внесе во прелистувач во претходниот чекор?