

Инструкция по разворачиванию ПО «Web-сервис Телемедицина»

свидетельство о государственной регистрации программы
для ЭВМ №2021611889

СОДЕРЖАНИЕ

1. Перечень сокращений и условных обозначений.....	2
2. Требования к серверным ресурсам	2
3. Порядок действий для разворачивания сервиса из дистрибутива.....	3
3.1. Общая информация.....	3
3.2. Пример конфигурации Nginx	3
3.3. Команды для разворачивания сервисов	4
4. Запуск системы.....	5
4.1. Запуск backend	5
4.2. Запуск frontend.....	7
4.3. Установка и запуск сервиса jitsi.....	7
4.4. Установка и запуск сервиса jibri	8
Приложение 1. Состав и архитектура сервиса ТМК	9

1. Перечень сокращений и условных обозначений

ПО	Программное обеспечение
ЭВМ	Электронно-вычислительная машина
ОС	Операционная система
СУБД	Система управления базами данных
ВКС	Видео конференц-связь
ТМК	Телемедицинская консультация
API	аббревиатура от английского application programming interface – программный интерфейс, описание способов взаимодействия одной компьютерной программы с другими

2. Требования к серверным ресурсам

Необходимы следующие сервера для установки программы для ЭВМ «Web-сервис Телемедицина»:

1. Сервер web-интерфейса под управлением ОС Windows Server и установленной СУБД MS SQL
2. Сервер организации проведения ВКС под управлением ОС Linux для разворачивания сервиса на базе Jitsi
3. Сервер потоковой передачи конференции под управлением ОС Linux для разворачивания сервиса на базе Jibri

Состав и архитектура сервиса с указанием минимальных требований к ресурсам приведены в Приложении 1 настоящей инструкции.

Минимальные характеристики серверов:

Jitsi: процессор 2 ядра 2.4 GHz, 4 Гб ОЗУ

Jibri: процессор 2 ядра 2.4 GHz, 2 Гб ОЗУ (на каждую активную единицу)

API ТМК: процессор 1 ядро 2.4 GHz, 2 Гб ОЗУ

Сайт ТМК: процессор 1 ядро 2.4 GHz, 2 Гб ОЗУ

В качестве обратного прокси-сервера для сайта и API ТМК используется Nginx.

Пример конфигурации Nginx приведён ниже.

Требуемые версии ПО:

Linux 4.14

Nginx 1.17

Docker 18

docker-compose 1.23

3. Порядок действий для разворачивания сервиса из дистрибутива

3.1. Общая информация

Дистрибутив сервиса ТМК состоит из 4 компонентов:

- Система Jitsi (онлайн конференции);
- Сервис Jibri (записи конференций) [опционально];
- API сервер ТМК (интеграция с внешними системами);
- Сайт ТМК.

Развёртывание ТМК должно происходить в порядке, перечисленном выше.

Инструкции развёртывания каждой из компонент описаны в соответствующих директориях.

Сайт ТМК Jitsi должен быть развёрнут на отдельном сервере; Jibri, сайт ВКС и API ТМК могут быть развёрнуты вместе.

Необходимо скопировать дистрибутивы на сервер и распаковать.

Важные функции сайта ТМК требуют исполнения в защищённом контексте.

(https://developer.mozilla.org/en-US/docs/Web/Security/Secure_Contexts)

Следовательно, доступ ко всем компонентам ТМК должен происходить по протоколу HTTPS.

API сервер ТМК и сайт ТМК не имеют встроенного механизма работы с HTTPS; доступ к ним

должен осуществляться через обратный прокси-сервер с терминованием TLS-соединений.

3.2. Пример конфигурации Nginx

=== Конфигурация Nginx

```
== /etc/nginx/nginx.conf
```

```
user www-data;  
worker_processes auto;  
pid /run/nginx.pid;  
include /etc/nginx/modules-enabled/*.conf;
```

```
events {  
    worker_connections 768;  
}
```

```
http {  
    proxy_buffer_size 16k;  
    proxy_busy_buffers_size 24k;  
    proxy_buffers 64 4k;
```

```

sendfile on;
tcp_nopush on;
tcp_nodelay on;
keepalive_timeout 65;
types_hash_max_size 2048;

server_names_hash_bucket_size 64;

include /etc/nginx/mime.types;
default_type application/octet-stream;

ssl_protocols TLSv1.2 TLSv1.3;
ssl_prefer_server_ciphers on;

gzip on;
include /etc/nginx/conf.d/*.conf;
}

== /etc/nginx/cors-all.conf

```

3.3. Команды для разворачивания сервисов

```

add_header 'Access-Control-Allow-Origin' '*' always;
add_header 'Access-Control-Allow-Methods' 'GET, POST, OPTIONS, DELETE, PUT, PATCH'
always;
add_header 'Access-Control-Allow-Headers' 'Authorization,DNT,X-CustomHeader,Keep-
Alive,User-Agent,X-Requested-With,If-Modified-Since,Cache-Control,Content-Type,Content-
Range,Range,CMI-Session-Id,Doctor-Call-Session-Id' always;

if ($request_method = 'OPTIONS') {
    add_header 'Access-Control-Max-Age' 1728000 always;
    add_header 'Content-Type' 'text/plain; charset=utf-8' always;
    add_header 'Content-Length' 0 always;
    return 204;
}

== /etc/nginx/conf.d/tmk-api.conf

server {
    server_name api.example.com;
    listen 443 ssl;
    client_max_body_size 200M;

    location / {
        include /etc/nginx/cors-all.conf;
        proxy_pass http://127.0.0.1:8080;
        proxy_http_version 1.1;
    }
}

```

```

        proxy_pass_request_headers on;
        proxy_set_header X-Forwarded-For $remote_addr;
        proxy_set_header Host $host;
    }

    ssl_certificate /path/to/cert.crt;
    ssl_certificate_key /path/to/cert.key;
}

== /etc/nginx/conf.d/tmk-site.conf

server {
    server_name tmk.example.com;
    listen 443 ssl http2;

    location / {
        proxy_pass http://127.0.0.1:4000;
        proxy_http_version 1.1;
        proxy_pass_request_headers on;
        proxy_set_header Host $http_host;
        proxy_set_header X-Forwarded-For $remote_addr;

    }
    ssl_certificate /path/to/cert.crt;
    ssl_certificate_key /path/to/cert.key;
}

```

4. Запуск системы

4.1. Запуск backend

Для запуска API сервера ТМК понадобятся:

1. Доступ в Интернет
2. Конфигурация Jitsi, а именно:
 - * Домен из PUBLIC_URL (пример: jitsi.example.com)
 - * JWT_APP_SECRET
3. Доменное имя сайта ТМК (пример: tmk.example.com)
4. Установленное ПО:
 - * docker
 - * docker-compose

Для запуска сервера ТМК достаточно выполнить скрипт `start.sh` с правами суперпользователя.

При первом запуске потребуется ввести нужные конфигурационные параметры. В дальнейшем изменить конфигурацию можно в файле env; для применения новой конфигурации нужно перезагрузить сервер.

(для перезагрузки сервера достаточно выполнить ./start.sh)

4.1.1. HTTPS

Для доступа к API серверу ТМК по протоколу HTTPS требуется использование стороннего ПО. В качестве обратного прокси-сервера. API сервер не требует особых заголовков запроса.

4.1.2. API

По-умолчанию API сервер принимает HTTP запросы по адресу 127.0.0.1 на порту 8080. Изменить привязанный адрес и порт можно в файле docker-compose.yml (services.tmk-api.ports).

Ниже представлен пример запроса на создание сессии ТМК.

```
curl 'http://<api_domain>:<api_port>/v1/abm/video/sessions/create/simple' \  
-H 'Content-Type: application/json' \  
-d '{  
  "companyLogoUrl": "file:///",  
  "companyName": "string",  
  "displayMode": "DEFAULT",  
  "doctorFullName": "<ФИО врача>",  
  "patientFullName": "<ФИО пациента>"  
}'
```

Ответ на запрос:

```
{  
  "result": {  
    "sessionId": 0,  
    "status": "OPEN",  
    "displayMode": "DEFAULT",  
    "doctorFullName": "<ФИО врача>",  
    "patientFullName": "<ФИО пациента>",  
    "room": "<room>",  
    "doctorJwt": "<doctor_jwt>",  
    "doctorLink": "https://<FRONT_DOMAIN>/session/0?room=<room>&jwt=<doctor_jwt>",  
    "patientJwt": "<patient_jwt>",  
    "patientLink": "https://<FRONT_DOMAIN>/session/0?room=<room>&jwt=<patient_jwt>"  
  }  
}
```

Если сайт ТМК и система Jitsi были настроены правильно, то сессию ТМК можно начать открыв в браузере ссылки из полей doctorLink и patientLink.

4.2. Запуск frontend

Для сборки сайта ТМК требуются:

1. Доступ в Интернет
2. Установленное ПО Docker
3. Домен системы Jitsi
4. Домен API-сервера ТМК

Для сборки необходимо выполнить скрипт: `./build.sh <docker_image_name> <jitsi_domain> <api_domain>`, где:

- * `<docker_image_name>` -- название (и опционально тег) собранного Docker-образа (пример: `tmk-front`)
- * `<jitsi_domain>` -- домен, на котором развёрнута система Jitsi (пример: `jitsi.example.com`)
- * `<api_domain>` -- домен, на котором развёрнут API-сервер ТМК (пример: `api.example.com`)

Собранный Docker-образ принимает подключения на порту 4000. Пример сборки и запуска:

```
sudo ./build.sh tmk-front:example jitsi.example.com api.example.com
sudo docker run -d -p 127.0.0.1:4444:4000 tmk-front:example
```

4.2.1. HTTPS

Для правильной работы подключение к сайту должно происходить по протоколу HTTPS. Для этого требуется использование стороннего ПО в качестве обратного прокси-сервера.

4.3. Установка и запуск сервиса jitsi

Для установки Jitsi требуется доступ в Интернет и установленное ПО:

- * `openssl`
- * `docker`
- * `docker-compose`

Jitsi использует 4 TCP и 1 UDP порт; все 5 портов должны быть доступны для внешних клиентов. Используемые порты могут быть изменены во время установки, за исключением порта 5222/TCP.

Также во время установки будет предложено использование Let's Encrypt для автоматического получения SSL сертификатов. Если использование Let's Encrypt неприемлемо, то после завершения установки нужно вручную заменить содержание файлов `cert.key` и `cert.crt` в директории `<путь_установки>/.cfg/web/keys` ключом сертификата и `fullchain`-сертификатом соответственно, после чего перезапустить Jitsi.

Если Jitsi разворачивается за обратным прокси-сервером, необходимо указать IP-адрес, по которому клиенты смогут подключаться к видеомосту, т.е. адрес обратного прокси-сервера.

Если же Jitsi имеет белый IP-адрес, то поле ввода публичного IP можно оставить пустым.

4.3.1. Порядок установки:

1. Выполнить скрипт (с правами суперпользователя): `./install.sh <путь_установки>`
2. Ввести требуемую конфигурацию

Вся конфигурация находится в файле `<путь_установки>/.env` и может быть изменена после установки.

Для применения изменений необходимо перезапустить Jitsi.

Для запуска, перезапуска и остановки Jitsi в директории установки прилагаются соответствующие скрипты.

4.4. Установка и запуск сервиса jibri

Для установки (и дальнейшей работы) Jibri требуются:

1. Доступ в Интернет
2. Конфигурация развёрнутой и настроенной системы Jitsi, а именно параметры:
 - * JIBRI_RECORDER_PASSWORD
 - * JIBRI_XMPP_PASSWORD
 - * PUBLIC_URL
3. Загруженный модуль ядра `snd_aloop`
4. Установленное ПО:
 - * `docker`
 - * `docker-compose`

ВАЖНО: если при установке Jitsi был указан публичный IP-адрес, то результатом DNS разрешения домена из `PUBLIC_URL` на сервере Jibri должен являться ранее указанный публичный IP адрес.

4.4.1. Порядок установки:

1. Выполнить скрипт (с правами суперпользователя): `./install.sh <путь_установки>`
2. Ввести конфигурационные данные (должны совпадать с конфигурацией Jitsi)
3. Ввести требуемое количество активных единиц Jibri

Вся конфигурация находится в файле `<путь_установки>/.env` и может быть изменена после установки. Для применения изменений необходимо перезапустить Jitsi.

Для запуска, перезапуска и остановки Jibri в директории установки прилагаются соответствующие скрипты.

Записи конференций хранятся в директории `<путь_установки>/recordings`.

Приложение 1. Состав и архитектура сервиса ТМК.



Сервер веб-сервиса ТМК

- ОС: Windows Server
- БД ТМК СУБД MS SQL
 - API отправки SMS



Сервер организации проведения видеоконференций

- ОС: Linux
- Jitsi сервис для проведения видеоконференций (ВКС)
 - API для управления ВКС
 - Запись и хранилище



Сервер потоковой передачи конференции

- ОС: Linux
- Jibri сервис для потоковой передачи конференции

Аппаратные требования		
Процессор : 2,4 ГГц ; Кол-во ядер 2; Оперативная память 4 Гб-8 Гб ; Дисковое пространство: 170 Гб-200 Гб	Процессор : 2.4 ГГц ; Кол-во ядер 2; Оперативная память 4 Гб-8 Гб ; Дисковое пространство: 200 Гб-600 Гб	Процессор : 2.4 ГГц ; Кол-во ядер 30; Оперативная память 22 Гб - 24 Гб ; Дисковое пространство: 170 Гб - 200 Гб
Программные требования		
<ul style="list-style-type: none"> • ОС Microsoft Windows Server 2012(R2) и выше (2016, 2019 и т.д.), 64bit • СУБД - MS SQL Server 2014 	ОС Ubuntu 20.04.3 LTS	ОС Ubuntu 20.04.3 LTS
Сетевые настройки		
	разрешить доступ к следующим ресурсам: tcp 80,443,4443,5222 udp 10000	