ООО "ГТС"

Специализированное программное обеспечение

Автоматизированная система формирования планового расписания движения трамваев

ИНСТРУКЦИЯ ПО УСТАНОВКЕ

Листов 14

Москва, 2025 г.

Подп. И дата
Инв. №
Взам. инв.
Подп. И дата
Инв. №

Содержание

Перечень используемых терминов, их определений и сокращений 3
1 Введение 5
2 Требования 5
3 Установка СПО 5
3.1 Развертывание проекта AS_FRT 5
3.2 Установка приложения gidwidget 11
4 Запуск клиентской части СПО АСФРТ 13
Лист регистрации изменений 14

Перечень используемых терминов, их определений и сокращений

Термин	Сокращение	Определение
Автоматизированное рабочее место	АРМ	Комплекс аппаратных и программных средств, обеспечивающий взаимодействие человека с компьютером и предоставляющий возможность ввода информации (через клавиатуру, компьютерную мышь) и её вывода на экран монитора, принтер, динамики или иные устройства вывода
Автоматизированная система формирования планового расписания движения трамваев	АСФРТ	Система, предназначенная для реализации автоматизированного процесса построения планового расписания движения, а также автоматического устранения конфликтов в расписании движения трамваев
График исполненного движения	ГИД	Графическое отображение фактического движения составов на текущий момент времени для отдельно взятого маршрута
Специализированное программное обеспечение	СПО	Совокупность взаимосвязанных программ, разработанных для решения задач конкретной предметной области или технологического производственного процесса и предусматривающих автоматизированное управление и функциональный контроль состояния комплекса автоматизированной системы диспетчерского управления
Docker		Открытая платформа для разработки, доставки и эксплуатации приложений

Internet Protocol	IP-адрес	Уникальный адрес, который идентифицирует устройство в Интернете или локальной сети
	NGINX	HTTP-сервер, обратный прокси сервер с поддержкой кеширования и балансировки нагрузки, TCP/UDP прокси-сервер, а также почтовый прокси-сервер
	NATS	Система для публикации и прослушивания сообщений на именованных каналах связи
Swarm		Функция Docker, которая предоставляет встроенные возможности оркестрации контейнеров, включая собственную кластеризацию узлов Docker и планирование рабочих нагрузок контейнеров
Протокол управления передачей	TCP	Фундаментальный протокол в наборе интернет-протоколов, в первую очередь предназначенный для надежной передачи данных по сетям. Он гарантирует, что пакеты данных доставляются точно и в правильном порядке
	VMware	Технология, которая позволяет создавать виртуальные машины на физических серверах и управлять ими, позволяя одновременно работать нескольким операционным системам на одной аппаратной платформе
	VXLAN	Технология виртуализации сети, предназначенная для решения проблем масштабируемости в больших средах облачных вычислений

1 Введение

Настоящий документ содержит сведения, необходимые для установки специализированного программного обеспечения Автоматизированной системы формирования планового расписания движения трамваев (далее – СПО АСФРТ).

2 Требования

Для установки СПО необходимы:

- оперативная память на АРМ: не менее 16 Гб;
- жёсткий диск на АРМ: не менее 20 Гб свободного места;
- оперативная память на сервере: не менее 16 Гб на каждую виртуальную машину;
- жёсткий диск на сервере: не менее 60 Гб свободного места на каждую виртуальную машину;
- установленная операционная система на АРМ Linux Ubuntu 20.04;
- серверная операционная система Linux Ubuntu 22.04.

3 Установка СПО

Для установки СПО необходимо:

1. Развернуть на сервере проект AS_FRT – проект, реализующий основную функциональность АСФРТ.
2. Установить приложение gidwidget на АРМ – приложение, реализующее визуальное отображение расписания в виде графика.

3.1 Развертывание проекта AS_FRT

Для развёртывания AS_FRT необходимо:

1. Обеспечить сервер версией Linux Ubuntu 22.04.
   После установки в системе есть пользователь - администратор – "master".
2. Добавить нового пользователя "tech":
   ```
   usedadd -G users -m -s /bin/bash tech
   ```

3. Задать новому пользователю "tech" пароль:

   ```
   passwd tech
   ```

   Данный пользователь, который будет управлять docker, без прав администратора системы.

4. Установить docker:

   ```
   apt install docker docker-compose
   ```

5. Добавить двух пользователей в группу docker:

   ```
   usermod master -aG docker
   usermod tech -aG docker
   ```

6. Настроить параметры docker:

   ```
   vim /etc/docker/daemon.json
   ```

7. Добавить ограничения на адреса сети (256 сетей /24)

   ```
   "default-address-pools": [
       { "base": "172.27.0.0/16", "size": 24 }
   ]
   ```

   Ограничение должно быть корректно написано с учётом синтаксиса json.

8. Добавить ограничения на логи

   ```
   "log-opts": {
       "max-size": "1g"
   }
   ```

   Ограничение должно быть корректно написано с учётом синтаксиса json.

9. Перезапустить сервис docker:

   ```
   service docker restart
   ```

10. На первом хосте из трёх включить swarm:

    ```
    docker swarm init --default-addr-pool 172.28.0.0/14
    --default-addr-pool-mask-length 22
    ```

    Эта команда ограничивает размер сеток в swarm: 256 /22 сетей.

11. Если виртуальные машины созданы в VMware и находятся на одном физическом сервере, то возможно возникнет проблема недоступности узлов кластера по tcp соединению, при этом icmp (ping) будет работать штатно. В этом случае проблема скорее всего связана с конфликтом портов VXLAN и VMware NSX's communication. Для её решения необходимо назначить другой порт (отличный от 4789) для docker swarm:
    `docker swarm init --data-path-port=3888`
    При выборе порта указать такой, который не входит в динамический диапазон портов, выдаваемых системой приложениям при установлении исходящих соединений. Для Ubuntu по умолчанию динамический диапазон: 32768-60999.
    Если в хосте настроено несколько сетевых карт и адресов, то docker выдаст предупреждение, что не указан адрес сети, в которой ему работать. Для решения этой проблемы необходимо добавить в предыдущую команду:
    `--advertise-addr <Требуемый IP-адрес>`
    После исполнения команды будет выдан текст, в котором написана:
    - команда для добавления в swarm нового member (токен авторизации member);
    - команда, при исполнении которой можно получить в текстовом виде новую команду для добавления в swarm нового manager (токен авторизации manager).
12. Получить команду для добавления в swarm нового manager и выполнить на 2 и 3 хосте:
    `docker swarm join --token <токен авторизации manager>`
13. После добавления всех хостов в swarm прописать им лейблы:
    `docker node update swarm-3a --label-add nodeNum=1`
    `docker node update swarm-3b --label-add nodeNum=2`

```
docker node update swarm-3c --label-add nodeNum=3
docker node update swarm-3a --label-add letter=a
docker node update swarm-3b --label-add letter=b
docker node update swarm-3c --label-add letter=c
```

14. Скопировать предоставленный исполнителем архив frt_deploy.zip с переносного носителя информации.
15. Распаковать архив с помощью команды:

    `unzip frt_deploy.zip`

    Из архива распаковывается каталог AS_FRT.
16. После распаковки в папке назначения должны находиться файлы:
    - конфигурационная инструкция docker-compose.yml;
    - скрипт manage.sh;
    - каталоги: env, map_server, updater;
    - конфигурационный файл .env;
    - конфигурационные файлы сервисов: frontend.conf, nats-conf.conf, nginx.conf, prest.toml, proxy-params.conf, stan.conf, WSServer_js, yas- conf.json;
    - сервисные приложения для реализации дополнительной функциональности для скрипта manage.sh: gum, lnav.
17. Проверить и при необходимости изменить содержание файла `.env`. Описание файла приведено ниже. Содержание файла проверяется и при необходимости исправляется с помощью любого текстового редактора, например, nano командой:

    `<имя пользователя на сервере>:<путь до файла> nano <имя файла>`

    При изменении файла в nano сохранить с помощью Ctrl+S, выйти из режима просмотра и редактирования – Ctrl+X.

18. Создать сеть docker, выполнив:

    ```
    ./manage.sh init
    ```

19. Запустить стак docker-контейнеров, выполнив:

    ```
    ./manage.sh start
    ```

20. Дождаться пока контейнеры загрузятся, для отслеживания статуса можно использовать команду:

    ```
    ./manage.sh status
    ```

21. При рабочем состоянии сервиса в столбцах DESIRED STATE и CURRENT STATE будет стоять Running.

В репозитории AS_FRT содержатся необходимые для развёртывания файлы:

1. `.env`

   Содержит необходимые для инициализации AS_FRT переменные окружения и их значения по умолчанию. Значения переменных в этом файле должны быть изменены настраивающим лицом на актуальные значения, с которыми AS_FRT будет инициализирован на промышленной эксплуатации.

   Пример конфигурационного файла .env:

   ```
   STACK=AS_FRT
   NETWORK=AS_FRT_NETWORK
   WSS_PORT=7878
   FRONT_PORT=3022
   ```

   Описание .env-файла приведено в таблице 1.

   Таблица 1 – Описание .env-файла

Параметр	Описание
STACK	Наименование кластера контейнера
NETWORK	Сеть кластера контейнера
WSS_PORT	Порт веб-сокет сервера
FRONT_PORT	Порт клиентского приложения

2. `docker-compose.yml`

   Содержит параметры запуска сервиса в swarm-режиме.

3. `manage.sh`

   Содержит общую логику старта, стопа сервисов в swarm-mode.

4. Каталог `env`

   Каталог с файлами переменных окружения для части сервисов АСФРТ.

5. Каталог `map_server`

   Каталог, содержащий файлы конфигурирования и сборки картографической подложки.

6. Каталог `updater`

   Каталог, содержащий конфигурацию сервиса обновления образов сервисов АСФРТ.

7. `frontend.conf`

   Конфигурационный файл клиентского веб-приложения.

8. `nats-conf.conf, stan.conf`

   Конфигурационные файлы брокера сообщений NATS.

9. `nginx.conf, proxy-params.conf`

   Конфигурационные файлы сервиса NGINX.

10. `prest.toml`

    Конфигурационный файл сервиса pREST.

11. `WSServer_js`

    Конфигурационный файл сервиса WSServer_JS.

12. `yas-conf.json`

    Конфигурационный файл для модуля построения расписания YaS.

13. `gum`

    Сервисное приложения для реализации интерактивных скриптов в manage.sh.

14. `lnav`

    Инструмент командной строки для просмотра системных журналов в manage.sh.

3.2 Установка приложения gidwidget

1. Обеспечить АРМ версией Linux Ubuntu 20.04.
2. В командной строке на ПК выполнить команду:
   `apt-get update && apt-get upgrade -y`
3. В командной строке на ПК выполнить команду:
   `apt-get install -q --assume-yes --no-install-recommends qtbase5-dev-tools build-essential qtbase5-dev libqt5websockets5-dev qtquickcontrols2-5-dev qtdeclarative5-dev libqt5sql5 libqt5widgets5 libqt5gui5 libqt5dbus5 qml-module-qtquick-controls qml-module-qtquick-controls2 qml-module-qtquick-dialogs qtbase5-private-dev libboost-dev libssl-dev libxi-dev libxtst-dev`
4. Создать директорию /home/<USERNAME>/AS_FRT_GID/ командой:
   `mkdir /home/<USERNAME>/AS_FRT_GID/`
5. Перейти в директорию /home/<USERNAME>/AS_FRT_GID/ командой:
   `cd /home/<USERNAME>/AS_FRT_GID/`
6. Скопировать в директорию /home/<USERNAME>/AS_FRT_GID/ предоставленный исполнителем архив gidwidget.zip с переносного носителя информации.
7. Распаковать архив с помощью команды:
   `unzip gidwidget.zip`
8. После распаковки в директории /home/<USERNAME>/AS_FRT_GID/ должны находиться файлы:
   - бинарный файл gidwidget;
   - конфигурационный файл gidwidget.ini.
9. Проверить и при необходимости изменить содержание блока SERVER1 файла gidwidget.ini, изменив параметры `ip` (внешний адрес виртуальной машины, на которой запущена серверная часть) и `port`

(порт прописан в файле docker-compose.yml для сервиса WSServer_JS).

Содержание файла проверяется и при необходимости исправляется с помощью любого текстового редактора, например, nano командой:

`<имя пользователя на сервере>:<путь до файла> nano <имя файла>`

При изменении файла в nano сохранить с помощью Ctrl+S, выйти из режима просмотра и редактирования – Ctrl+X.

10. Создать ярлык "ГИД_АС_ФРТ":

`nano /home/<USERNAME>/Desktop/ГИД_АС_ФРТ.desktop`

Откроется редактор nano с файлом ГИД_АС_ФРТ.desktop, скопировать туда содержимое, приведенное ниже и нажать Ctrl+X.

```
[Desktop Entry]
Comment[en_US]=
Comment=
Comment[ru_RU]=
Exec=/home/<USERNAME>/AS_FRT_GID/gidwidget
GenericName[en_US]=
GenericName=
GenericName[ru_RU]=
Icon=help-about
MimeType=
Name=ГИД_АС_ФРТ
Name[ru_RU]=ГИД
Path=/home/<USERNAME>/AS_FRT_GID
StartupNotify=true
Terminal=false
TerminalOptions=
Type=Application
X-DBUS-ServiceName=
X-DBUS-StartupType=none
X-KDE-SubstituteUID=false
X-KDE-Username=
```

4 Запуск клиентской части СПО АСФРТ

Для запуска клиентской части СПО АСФРТ необходимо:

1. Открыть браузер. Рекомендуется использовать Google Chrome.
2. В строке адреса ввести:
   `http:// <IP-адрес виртуальной машины, где развернута серверная часть>:<Порт клиентского приложения>`
3. Выполнить авторизацию в АСФРТ.

Для запуска gidwidget необходимо:

1. Найти на рабочем столе АРМ ярлык с надписью "ГИД_АС_ФРТ".
2. Дважды нажать по ярлыку, после этого на мониторе отобразится окно загрузки графического интерфейса ГИД.

Лист регистрации изменений

Изм.	Номера листов (страниц)				Всего листов (страниц) в документ	№ документа	Подп.	Дата
	изменён-ных	заменён-ных	новых	аннули-рован-ных				

Подп. И дата | Инв. № | Взам. инв. | Подп. И дата | Инв. №