

**МНОГОФУНКЦИОНАЛЬНАЯ ТЕЛЕМАТИЧЕСКАЯ
ПЛАТФОРМА
«ТиМаС»**

ИНСТРУКЦИЯ ПО УСТАНОВКЕ ДИСТРИБУТИВА

1 Введение

Телематическая Интеллектуальная Мониторинговая Автоматизированная Система (в дальнейшем будет использоваться термин ТиМаС) представляет программно – технический комплекс, предназначенный для мониторинга подвижных (мобильных) объектов.

Под подвижным объектом мониторинга понимается транспортное средство, оснащённое системой контроля МТ (мобильный терминал) позволяющей контролировать состояние контактных датчиков, параметрических датчиков и местоположение (а также скорость движения, пройденное расстояние и др.) транспортного средства (ТС) и передавать информацию по каналу GSM 3/4G (SMS, Data, Internet, др.). Существует возможность удалённого управления оборудованием, подключенным к выходам (реле), например, блокировать двигатель в случае угона.

В состав основных задач «ТиМаС» входит:

- Осуществлять сбор и хранение информации, поступающей от объектов мониторинга. В качестве поступающей информации может быть информация об изменении местоположения объектов или изменении состояния объектов (состояния датчиков, установленных на объектах).
- Выполнять запросы диспетчера о последнем состоянии объектов, а также за любой период времени.
- На основе полученных данных производить анализ ситуации и выявлять отклонения от нормального функционирования объектов (срабатывание аварийных датчиков, выезд транспортных средств из зоны) и сообщать об этом диспетчеру.
- Отображать информацию о последнем местоположении объектов (или за любой период) на электронной карте.
- Осуществлять тестирование как всего программно – технического комплекса, так и МТ, установленных на объектах.

Телематическая Интеллектуальная Мониторинговая Автоматизированная «ТиМаС» может быть использована для решения широкого круга задач:

- Централизация управления перевозками грузов, в том числе ценных и опасных.
- Получение информации о состоянии транспорта и перевозимых грузов в реальном масштабе времени, через определенные промежутки времени или после прибытия в парк.
- Поиск транспортных средств по показаниям их навигационных датчиков.
- Оперативная оптимизация работы патрульных и спасательных служб.
- Осуществление централизованного оперативного сбора информации о состоянии окружающей среды, данных инженерных изысканий при помощи подвижных лабораторий на автомобилях, вертолетах, самолетах.
- Отображение массива данных в режиме реального времени, помогут не только сократить перерасходы ГСМ, но и избавиться от простоев, перепробегов, нецелевого использования техники.

2 Принцип построения

Структурная схема взаимодействия программного обеспечения на сервере «ТиМаС» изображена на рисунке 1.

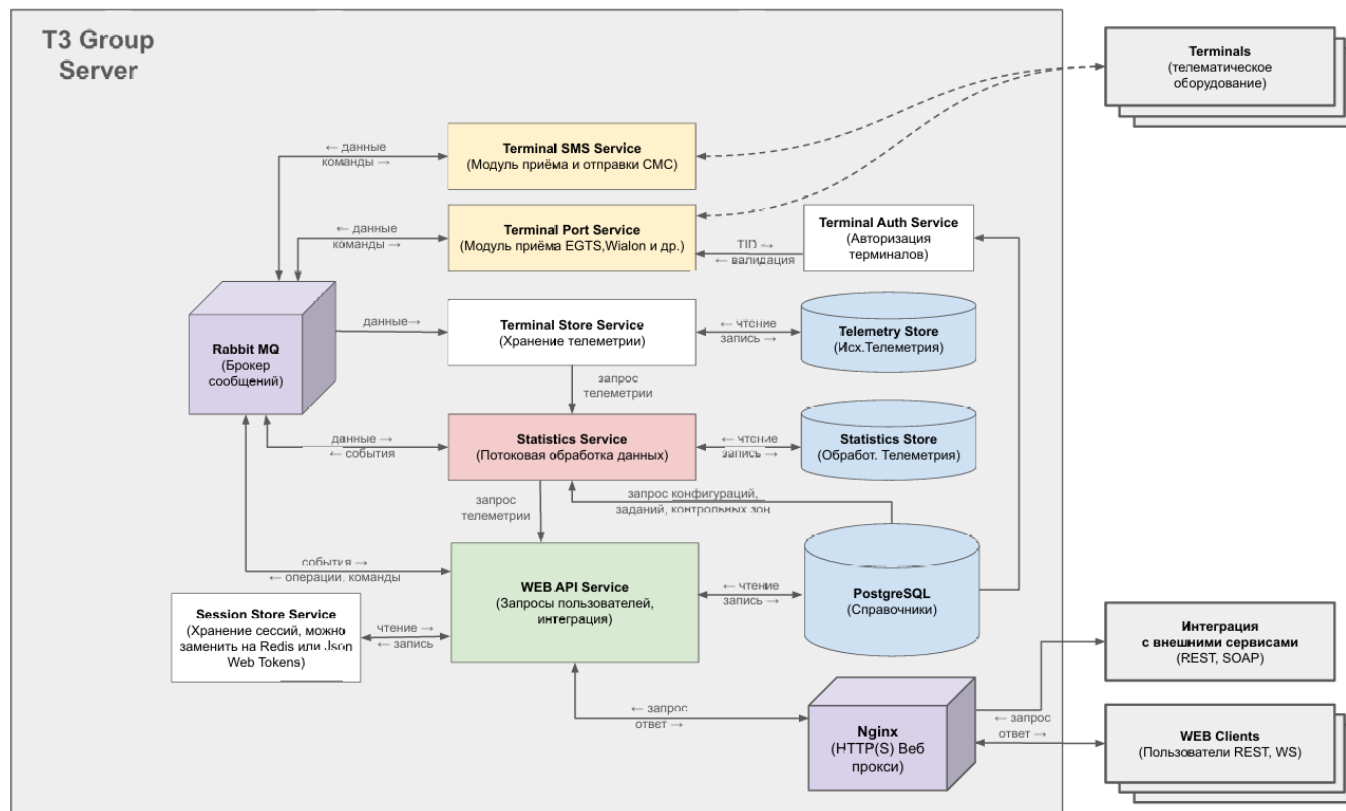


Рис.1

3 Программно-технические требования

Для нормально функционирования сервера «ТиМаС» необходимы следующие программно – технические средства:

- Сервер или виртуальная машина с процессором не ниже E5-2650v2 Процессор Intel Xeon Processor E5-2650 v2, ОЗУ не менее 16 Гбайт, место на диске SAS от 1.2 Тбайт в зависимости от количества устанавливаемых электронных карт и планируемого размера базы данных.
- ОС Ubuntu 24.04.4 LTS долгосрочная поддержка.
- При использовании СУБД PostgreSQL 16 следует руководствоваться минимальными требованиями к этим системам.

4 Установка ПО ТиМаС на Linux

4.1 Подготовка базового ПО на Ubuntu 24.04

Во-первых, необходимо обновить базовое ПО, нужно ввести такую команду в командной строке и нажать Enter:

```
sudo apt update && sudo apt upgrade -y
```

Во-вторых, обновление базовых утилит, нужно ввести такую команду в командной строке и нажать Enter:

```
sudo apt install -y curl wget gnupg software-properties-common ca-certificates
```

5 Установка базового ПО

5.1 Установка СУБД PostgreSQL 16

Добавление репозитория PostgreSQL, нужно ввести следующие команды в командной строке и нажать Enter:

```
sudo sh -c 'echo "deb https://apt.postgresql.org/pub/repos/apt $(lsb_release -cs)-pgdg  
main" > /etc/apt/sources.list.d/pgdg.list'  
wget -qO- https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo tee  
/etc/apt/keys.d/pgdg.asc  
sudo apt update
```

Установка СУБД PostgreSQL, нужно ввести такую команду в командной строке и нажать Enter:

```
sudo apt install -y postgresql-16 postgresql-client-16
```

Настройка СУБД PostgreSQL, нужно ввести следующие команды в командной строке и нажать Enter:

```
sudo systemctl start postgresql  
sudo systemctl enable postgresql
```

Создание базы данных и пользователя СУБД PostgreSQL (пример), нужно ввести такую команду в командной строке и нажать Enter:

```
sudo -u postgres psql <<EOF
CREATE USER timas_admin WITH PASSWORD 'StrongPassword123!';
CREATE DATABASE timas_db OWNER timas_admin;
ALTER USER timas_admin WITH SUPERUSER;
\q
EOF
```

Настройка аутентификации (опционально, для удаленного доступа) СУБД PostgreSQL (пример), нужно ввести такую команду в командной строке и нажать Enter:

```
sudo nano /etc/postgresql/16/main/pg_hba.conf
```

Добавить строку для локального доступа: local all timas_admin md5 СУБД PostgreSQL, нужно ввести такую команду в командной строке и нажать Enter:

```
sudo systemctl restart postgresql
```

5.2 Установка брокера сообщений RabbitMQ

Установка Erlang (зависимость RabbitMQ), нужно ввести такую команду в командной строке и нажать Enter:

```
curl -fsSL https://packages.erlang-solutions.com/ubuntu/erlang_solutions.asc | sudo gpg --dearmor -o
/usr/share/keyrings/erlang.gpg
echo "deb [signed-by=/usr/share/keyrings/erlang.gpg] https://packages.erlang-solutions.com/ubuntu
$(lsb_release -cs) contrib" | sudo tee /etc/apt/sources.list.d/erlang.list
```

Установка RabbitMQ, нужно ввести следующие команды в командной строке и нажать Enter:

```
curl -fsSL https://github.com/rabbitmq/signing-keys/releases/download/2.0/rabbitmq-release-signing-key.asc | sudo gpg --dearmor -o /usr/share/keyrings/rabbitmq.gpg
echo "deb [signed-by=/usr/share/keyrings/rabbitmq.gpg] https://packagecloud.io/rabbitmq/rabbitmq-server/ubuntu/ $(lsb_release -cs) main" | sudo tee /etc/apt/sources.list.d/rabbitmq.list

sudo apt update
sudo apt install -y erlang-base erlang-crypto rabbitmq-server
```

Запуск и настройка RabbitMQ, нужно ввести такую команду в командной строке и нажать Enter:

```
sudo systemctl start rabbitmq-server
sudo systemctl enable rabbitmq-server
```

Создание пользователя и виртуального хоста timas, нужно ввести следующие команды в командной строке и нажать Enter:

```
sudo rabbitmqctl add_user timas_user timas_rabbitmq_pass
sudo rabbitmqctl add_vhost timas_vhost
sudo rabbitmqctl set_permissions -p timas_vhost timas_user ".*" ".*" ".*"
sudo rabbitmqctl set_user_tags timas_user administrator
```

Включение плагина управления, нужно ввести такую команду в командной строке и нажать Enter:

```
sudo rabbitmq-plugins enable rabbitmq_management
```

5.3 Установка веб-сервера Nginx

Установка веб-сервера Nginx, нужно ввести такую команду в командной строке и нажать Enter:

```
sudo apt install -y nginx
```

Запуск и включение автозагрузки веб-сервера Nginx, нужно ввести следующие команды в командной строке и нажать Enter:

```
sudo systemctl start nginx  
sudo systemctl enable nginx
```

Настройка брандмауэра (если включен), нужно ввести следующие команды в командной строке и нажать Enter:

```
sudo ufw allow 'Nginx Full'  
sudo ufw allow ssh  
sudo ufw --force enable
```

6 Установка серверного ПО ТиМаС

6.1 Установка GIT

Установка GIT, нужно ввести такую команду в командной строке и нажать Enter:

```
sudo apt install -y git
```

6.2 Установка NodeJS + NPM (через NodeSource)

Установка Node.js 20.x (LTS), нужно ввести такую команду в командной строке и нажать Enter:

```
curl -fsSL https://deb.nodesource.com/setup_20.x | sudo -E bash -  
sudo apt install -y nodejs
```

Проверка версий, нужно ввести такую команду в командной строке и нажать Enter:

```
node --version  
npm --version
```

6.3 PM2 (Node Process Manager)

Установка PM2, нужно ввести такую команду в командной строке и нажать Enter:

```
sudo npm install -g pm2
```

Настройка автозагрузки PM2, нужно ввести такую команду в командной строке и нажать Enter:

```
pm2 startup
```

Выполните команду, которую предложит система

Сохранение текущей конфигурации PM2:

```
pm2 save
```

6.4 Установка ПО Timas Web

Создание рабочей директории ПО Timas:

```
sudo mkdir -p /opt/timas  
sudo chown -R $USER:$USER /opt/timas
```

Скопировать репозиторий timas-web из локального дистрибутива или через GIT:

```
cd /opt/timas  
git clone <timas_web_repository_url> web  
cd web
```

Установка зависимостей:

```
npm install
```

Сборка проекта:

```
npm run build
```

Конфигурирование, нужно скопировать файлы конфигурации из примеров, открыть в текстовом редакторе и исправить параметры доступа к БД и RabbitMQ:

```
cp ./config/mon.example.js ./config/mon.js  
cp ./config/storage.example.js ./config/storage.js  
cp ./config/notify.example.js ./config/notify.js
```

Запустите скрипт инициализации БД через команду, она создаст необходимую структуру таблиц и учётную запись администратора:

```
npm run init
```

Запуск Timas Web сервисов через PM2:

```
pm2 start pm2.config.js  
pm2 save
```

6.5 Установка ПО Timas Term

Скопировать репозиторий timas-term из локального дистрибутива или через GIT:

```
cd /opt/timas  
git clone <timas_term_repository_url> term  
cd term
```

Установка зависимостей:

```
npm install
```

Конфигурирование, нужно скопировать файлы конфигурации из примеров, открыть в текстовом редакторе и исправить параметры доступа к БД и RabbitMQ:

```
cp ./config/port.example.js ./config/port.js  
cp ./config/rbmq.example.js ./config/rbmq.js  
cp ./config/auth.example.js ./config/auth.js  
cp ./config/store.example.js ./config/store.js
```

Запуск Timas Term сервисов через PM2:

```
pm2 start pm2.config.js  
pm2 save
```

6.6 Установка ПО Timas Stat

Скопировать репозиторий timas-stat из локального дистрибутива или через GIT:

```
cd /opt/timas  
git clone <timas_stat_repository_url> stat  
cd stat
```

Установка зависимостей:

```
sudo apt install -y install \  
build-essential gdb \  
libz-dev \  
libboost-dev \  
libboost-system-dev \  
libboost-filesystem-dev \  
libboost-date-time-dev \  
libasio-dev \  
libtbb-dev \  
libzeroc-ice-dev \  
librabbitmq-dev
```

Сборка проекта:

```
make release TARGET=monstat
```

Настройте параметры сервиса статистики:

```
cp ./config/monstat.example.js ./config/monstat.js  
nano ./config/monstat.js
```

Запуск сервиса:

```
sudo touch /etc/systemd/system/monstat.service  
sudo nano /etc/systemd/system/monstat.service  
sudo chmod 644 /etc/systemd/system/monstat.service  
sudo systemctl daemon-reload  
sudo systemctl start monstat.service  
sudo systemctl enable monstat.service  
sudo journalctl -f -u monstat.service
```

6.7 Настройка Nginx

Для объединения всех сервисов на 80 порту или для настройки защищённого подключения SSL на 443 порту рекомендуется использовать Nginx сервер. В начале необходимо создать и отредактировать конфигурацию для Timas:

```
sudo cp /opt/timas/web/config/nginx.js /etc/nginx/sites-available/timas  
sudo nano /etc/nginx/sites-available/timas
```

Активация конфигурации:

```
sudo ln -s /etc/nginx/sites-available/timas /etc/nginx/sites-enabled/  
sudo nginx -t  
sudo nginx -s reload
```

6.8 Проверка работоспособности

Проверка статуса всех сервисов:

```
sudo systemctl status postgresql rabbitmq-server nginx  
sudo pm2 status
```

Проверка сетевых портов:

Проверка логов:

```
sudo pm2 logs  
sudo journalctl -f -u monstat.service  
sudo journalctl -f -u postgresql
```

6.9 Команды управления

Управление PM2 процессами

```
sudo pm2 list      # Список процессов  
sudo pm2 logs term  # Логи конкретного приложения  
sudo pm2 restart all # Перезапуск всех приложений  
sudo pm2 monit      # Мониторинг в реальном времени
```

Управление базами данных

```
sudo -u postgres psql -d timas_db
```

Управление RabbitMQ

```
sudo rabbitmqctl list_users  
sudo rabbitmqctl list_vhosts
```

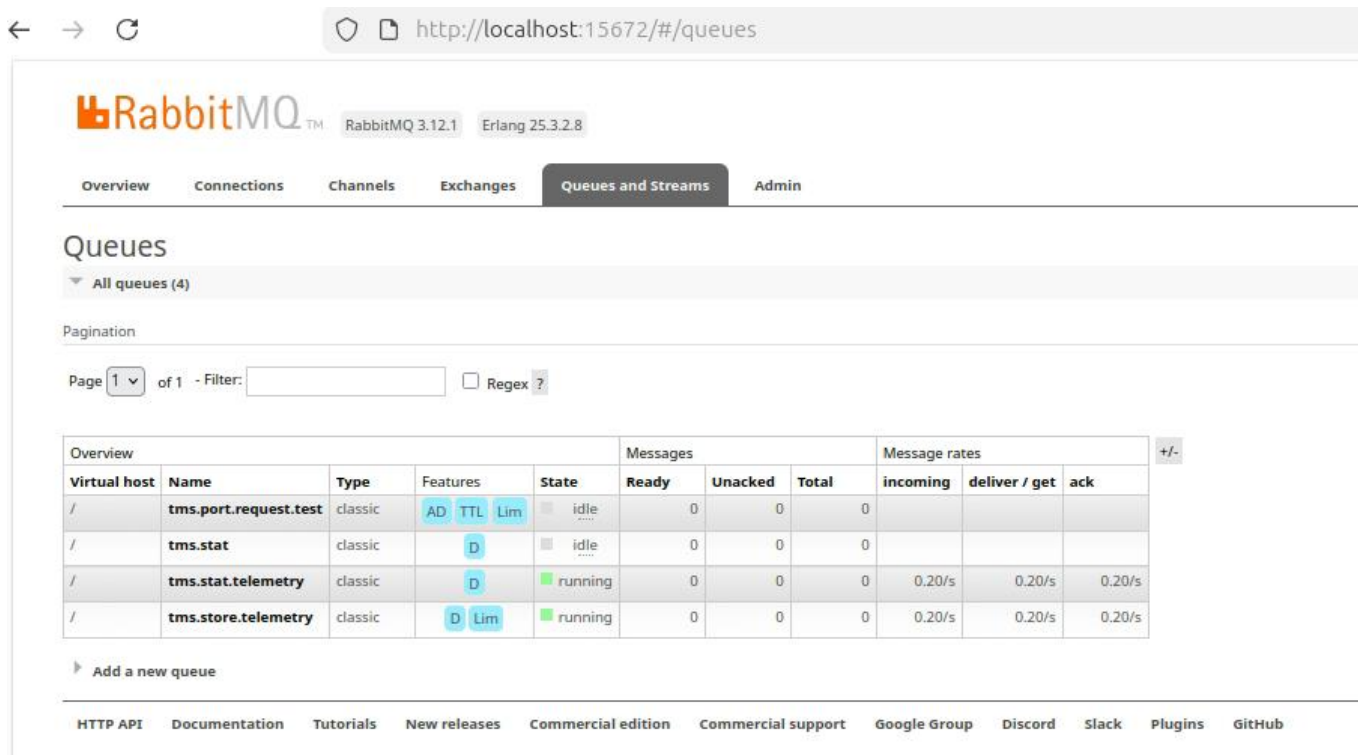
Перезагрузка системы

```
sudo pm2 save  
sudo reboot
```

Для доступа к веб-интерфейсу RabbitMQ необходимо активировать плагин управления

```
sudo rabbitmq-plugins enable rabbitmq_management
```

После этого можно открыть браузер и перейти по адресу <http://localhost:15672>



RabbitMQ 3.12.1 Erlang 25.3.2.8

Overview Connections Channels Exchanges **Queues and Streams** Admin

Queues

All queues (4)

Pagination

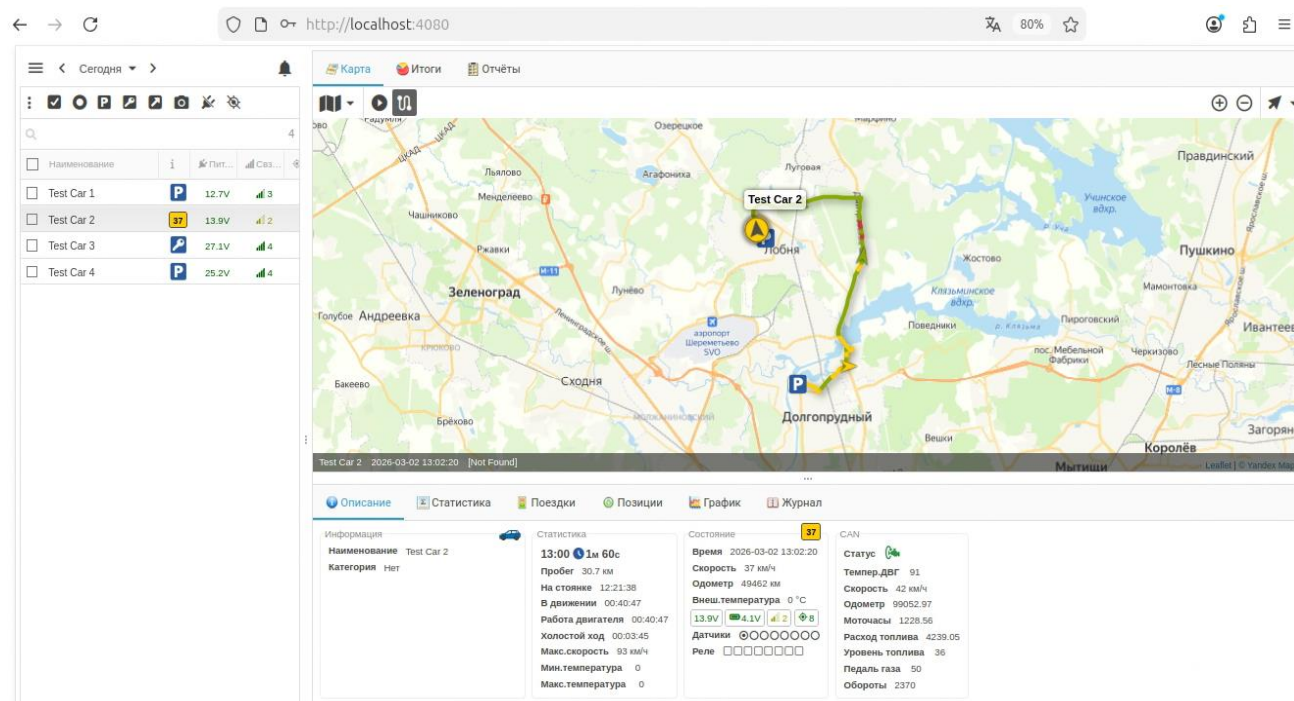
Page 1 of 1 - Filter: ☐ Regex ?

Overview					Messages			Message rates		
Virtual host	Name	Type	Features	State	Ready	Unacked	Total	incoming	deliver / get	ack
/	tms.port.request.test	classic	AD TTL Lim	idle	0	0	0			
/	tms.stat	classic	D	idle	0	0	0			
/	tms.stat.telemetry	classic	D	running	0	0	0	0.20/s	0.20/s	0.20/s
/	tms.store.telemetry	classic	D Lim	running	0	0	0	0.20/s	0.20/s	0.20/s

[Add a new queue](#)

[HTTP API](#)
[Documentation](#)
[Tutorials](#)
[New releases](#)
[Commercial edition](#)
[Commercial support](#)
[Google Group](#)
[Discord](#)
[Slack](#)
[Plugins](#)
[GitHub](#)

Для доступа к веб-интерфейсу Timas WEB откройте браузер и перейдите по адресу <http://localhost:4080>



http://localhost:4080

Карта Итоги Отчёты

Test Car 2

13:00 1m 60s

Пробег 30.7 км

На движение 12:21:38

В движении 00:40:47

Работа двигателя 00:40:47

Холостый ход 00:03:45

Макс. скорость 93 км/ч

Мин. температура 0

Макс. температура 0

Время 2026-03-02 13:02:20

Скорость 37 км/ч

Одометр 49462 км

Внеш. температура 0 °C

13.9V 4.1V 4.2 4.8

Датчики 00000000

Реле 00000000

Статус 37

Темпер. ДВГ 91

Скорость 42 км/ч

Одометр 99052.97

Моточасы 1228.56

Расход топлива 4239.05

Уровень топлива 36

Педаль газа 50

Обороты 2370

Заметки _____

Контактная информация:

ООО «Триангл»

Адрес: Москва, Бизнес-парк Румянцево, стр.2, корпус "Г", офис 836/2Г

Телефон: +7 (495) 846-78-17

E-mail: INFO@T3GROUP.RU

<https://t3group.ru>