

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ
«БАЙТЭРГ»

Утвержден
RU.БРУА.00065-01 34 01-ЛУ

СИСТЕМА ПО УПРАВЛЕНИЮ МОБИЛЬНЫМИ УСТРОЙСТВАМИ ОКО
«MDM ОКО»

Руководство оператора

RU.БРУА.00065-01 34 01

| | | | | |
|---------------|--------------|--------------|---------------|--------------|
| Инов. № подл. | Подп. и дата | Взам. инв. № | Инов. № дубл. | Подп. и дата |
| | | | | |

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

АННОТАЦИЯ

В данном программном документе приведено руководство оператора по применению и эксплуатации RU.БРУА.00065-01 Система по управлению мобильными устройствами ОКО «MDM ОКО» (далее по тексту - СПУ «MDM ОКО» или СПУ).

Настоящее руководство оператора (далее по тексту - руководство) состоит из следующих разделов:

- назначение и условия применения СПУ «MDM ОКО»;
- системные требования;
- процедура установки серверной части СПУ «MDM ОКО»;
- первоначальная настройка СПУ «MDM ОКО»;
- установка агента СПУ «MDM ОКО» на мобильное устройство;
- дополнительные операции.

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

СОДЕРЖАНИЕ

| | |
|--|----|
| Аннотация | 2 |
| 1. Назначение и условия применения СПУ «МДМОКО» | 5 |
| 2. Системные требования | 7 |
| 2.1. Аппаратные требования к серверу | 7 |
| 2.2. Программные требования | 7 |
| 2.3. Требования к сети и портам..... | 8 |
| 3. Установка серверной части из Debian-пакета | 9 |
| 3.1. Подготовка операционной системы..... | 9 |
| 3.2. Подключение репозитория и установка пакета | 10 |
| 3.3. Интерактивный мастер установки..... | 11 |
| 3.4. Проверка работоспособности после установки | 12 |
| 4. Установка серверной части с использованием Docker-контейнера..... | 13 |
| 4.1. Установка зависимостей..... | 13 |
| 4.2. Загрузка и распаковка дистрибутива | 13 |
| 4.3. Настройка конфигурации | 14 |
| 4.4. Установка сертификатов HTTPS | 16 |
| 4.5. Запуск контейнера и проверка | 17 |
| 5. Настройка HTTPS для веб-панели в Docker-контейнере..... | 19 |
| 6. Настройка локального сервера картографии | 20 |
| 6.1. Установка и первоначальный запуск | 20 |
| 6.2. Настройка автозапуска в виде системного сервиса..... | 21 |
| 7. Начальная настройка СПУ после установки..... | 23 |
| 7.1. Первый вход в веб-панель..... | 23 |
| 7.2. Первоначальная настройка конфигурации мобильных устройств..... | 23 |
| 7.3. Управление пользователями и ролями | 24 |
| 8. Установка агента СПУ на мобильное устройство | 25 |
| 8.1. Ручная установка APK..... | 25 |

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

| | |
|---|----|
| 8.2. Установка через QR-код..... | 26 |
| 8.3. Настройка прав администратора мобильного устройства (Device Owner)..... | 27 |
| 9. Дополнительные операции..... | 28 |
| 9.1. Импорт и экспорт конфигураций | 28 |
| 9.2. Добавление файлов обновлений программного обеспечения мобильных устройств..... | 29 |
| 9.3. Регистрация ключа картографического сервиса MapTiler | 30 |
| 9.4. Обновление серверной части СПУ..... | 30 |

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

1. НАЗНАЧЕНИЕ И УСЛОВИЯ ПРИМЕНЕНИЯ

СПУ «MDM ОКО»

1.1. СПУ «MDM ОКО» предназначена для обеспечения централизованного управления, мониторинга и конфигурирования мобильных устройств линейки ОКО на базе операционной системы Android (далее по тексту - мобильные устройства). СПУ предназначена для развертывания в корпоративных сетях, в том числе в закрытых сетях без доступа к сети Интернет.

1.2. Основные функции СПУ «MDM ОКО»:

- централизованное управление парком мобильных устройств через веб-панель администратора;
- удаленная установка, обновление и удаление приложений без подтверждения пользователя;
- управление конфигурациями мобильных устройств (профилями Wi-Fi, ограничениями, политикой безопасности);
- мониторинг состояния мобильных устройств (заряд аккумуляторной батареи, наличие в сети, журналы);
- удаленные команды: «Перезагрузка», «Блокировка», «Сброс к заводским настройкам»;
- управление аппаратными функциями: Wi-Fi, Bluetooth, GPS, яркость, громкость;
- доставку push-уведомлений и сообщений пейджера;
- поддержку режимов «COBO» и «COSU»;
- многофакторную аутентификацию (TOTP/MFA) для веб-панели;
- мультитенантный режим (multi-tenant) с разграничением данных организаций;
- развертывание агента с помощью QR-кода (Android 7.0 и выше).

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

1.3. Состав инфраструктуры для установки и эксплуатации СПУ «MDM ОКО»

1.3.1. Для установки и эксплуатации серверной части СПУ «MDM ОКО» должны быть обеспечены следующие параметры:

- сервер или виртуальная машина под управлением ОС Linux (Debian 11/12 или Ubuntu 20/22/24);
- сетевое подключение: доступность сервера по IP-адресу или DNS-имени с управляемых мобильных устройств;
- рабочая станция администратора с браузером для доступа к веб-панели (Google Chrome, Mozilla Firefox);
- кабель с USB Type-C для первоначальной установки агента СПУ на мобильное устройство (при необходимости).

1.3.2. Для установки и эксплуатации агента СПУ «MDM ОКО» на мобильные устройства должны быть обеспечены следующие параметры:

- мобильное устройство под управлением Android версии 7.0 и выше (для режима «Device Owner»);
- APK-файл агента СПУ «MDM ОКО», доступный в веб-панели или предоставленный ООО «Байтэрг».

Примечание - Для установки агента СПУ «MDM ОКО» через QR-код, должен быть обеспечен доступ мобильного устройства по сети к серверу с расположенным агентом СПУ «MDM ОКО».

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

2. СИСТЕМНЫЕ ТРЕБОВАНИЯ

2.1. Аппаратные требования к серверу

2.1.1. Выбор конфигурации сервера зависит от предполагаемого количества управляемых мобильных устройств. Рекомендуемые аппаратные конфигурации приведены в таблице 2.1.

Таблица 2.1 - Рекомендуемые аппаратные конфигурации сервера СПУ

| Количество мобильных устройств, шт. | Объем оперативного запоминающего устройства, ГБ | Количество ядер процессора, шт. | Объем накопителя информации (SSD), ГБ |
|-------------------------------------|---|---------------------------------|---------------------------------------|
| 1 От 1 до 200 | 4 | 2 | 50 |
| 2 От 200 до 1000 | 8 | 2 | 50 |
| 3 От 1000 до 5000 | 16 | 4 | 200 |
| 4 От 5000 до 20000 | 32 | 6 | 200 (отдельный сервер базы данных) |

2.1.2. Минимальные аппаратные требования к серверу должны соответствовать строке 1 таблицы 2.1 настоящего руководства.

2.2. Программные требования

2.2.1. Поддерживаемые операционные системы сервера:

- Debian 11 (Bullseye) - рекомендуется;
- Debian 12 (Bookworm);
- Ubuntu 20.04 LTS;
- Ubuntu 22.04 LTS;
- Ubuntu 24.04 LTS (требуется дополнительная настройка репозитория для Tomcat 9).

2.2.2. Программные компоненты, устанавливаемые автоматически в составе пакета:

- Apache Tomcat 9 - сервер приложений;
- PostgreSQL - система управления базами данных;
- Java Runtime Environment (JRE) - среда выполнения Java-приложений.

Примечание - Tomcat 10 не поддерживается. На Ubuntu 24.04 для доступности пакета tomcat9 необходимо подключить репозиторий Ubuntu Jammy (см. п. 3.1).

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

2.2.3. Для установки с использованием Docker-контейнера дополнительно требуется Docker Engine и docker-compose (версия 1, исполняемый файл /usr/local/bin/docker-compose).

2.3. Требования к сети и портам

2.3.1. Для корректной работы СПУ на сервере должны быть открыты порты, перечисленные в таблице 2.2.

Таблица 2.2 - Требуемые сетевые порты СПУ

| Протокол/порт | Назначение | Обязательность |
|---------------|---|----------------|
| TCP/443 | HTTPS-доступ к веб-панели (по умолчанию) | Обязательный |
| UDP/443 | Инициализация агента при регистрации по QR-коду | Обязательный |
| TCP/8080 | HTTP-доступ к веб-панели (по умолчанию) | Обязательный |
| TCP/9090 | Локальный сервер карт (tileserver), при наличии | Опциональный |
| TCP/31000 | MQTT-брокер (по умолчанию 31000) | Опциональный |

Примечание - При использовании HTTP (без HTTPS) вместо порта 443 используется порт 8080.

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

3. УСТАНОВКА СЕРВЕРНОЙ ЧАСТИ ИЗ DEBIAN-ПАКЕТА

Данный способ установки предназначен для развертывания СПУ «MDM ОКО» непосредственно на основной операционной системе Debian/Ubuntu без использования контейнеризации, и является рекомендуемым способом для промышленной эксплуатации на выделенном сервере.

3.1. Подготовка операционной системы

3.1.1. Для Debian 12 (Bookworm) необходимо добавить репозиторий Bullseye с целью получения пакета Tomcat 9:

```
echo "deb http://deb.debian.org/debian bullseye main" | sudo tee
/etc/apt/sources.list.d/bullseye.list
sudo tee /etc/apt/preferences.d/bullseye-pin <<EOF
Package: *
Pin: release n=bullseye
Pin-Priority: 100
Package: tomcat9*
Pin: release n=bullseye
Pin-Priority: 990
EOF
sudo apt update
```

3.1.2. Для Ubuntu 24.04 необходимо временно подключить репозиторий Jammy:

```
sudo nano /etc/apt/sources.list.d/jammy.list
```

Добавить строку:

```
deb [arch=amd64] http://archive.ubuntu.com/ubuntu jammy main universe
sudo apt update
```

Примечание - После завершения установки СПУ следует удалить файл репозитория и обновить индексы: `sudo rm /etc/apt/sources.list.d/jammy.list && sudo apt update`.

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

3.1.3. Убедиться, что файл `/etc/apt/sources.list` содержит корректные строки для используемого дистрибутива. Пример для Debian 11:

```
deb https://deb.debian.org/debian bullseye main contrib non-free non-free-
firmware
deb-src https://deb.debian.org/debian bullseye main non-free-firmware

deb https://security.debian.org/debian-security bullseye-security main non-
free-firmware
deb-src https://security.debian.org/debian-security bullseye-security main
non-free-firmware

deb https://deb.debian.org/debian bullseye-updates main non-free-firmware
deb-src https://deb.debian.org/debian bullseye-updates main non-free-firmware
```

3.2. Подключение репозитория и установка пакета

3.2.1. Способ 1. Установка внутреннего репозитория организации (при наличии собственного репозитория в организации):

```
sudo wget -O /etc/apt/trusted.gpg.d/okomdm.gpg http://repo-ip-
addr/deb/public.key

echo "deb [signed-by=/etc/apt/trusted.gpg.d/okomdm.gpg] http://repo-ip-
addr/deb bullseye main" | sudo tee /etc/apt/sources.list.d/okomdm.list

sudo apt-get update && sudo apt-get install oko-mdm-server -y
```

3.2.2. Способ 2. Установка из загруженного deb-пакета:

```
sudo apt-get install -y ./oko-mdm-server_6.0.0_amd64.deb
```

Примечание - Актуальное имя файла пакета и контрольную сумму необходимо уточнять у разработчика СПУ.

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

3.3. Интерактивный мастер установки

3.3.1. В процессе установки запустится интерактивный мастер, последовательно запрашивающий параметры системы. Описание запросов приведено в таблице 3.1.

Таблица 3.1 - Параметры интерактивного мастера установки СПУ

| Запрос мастера | Описание | Пример значения |
|-----------------------------------|--|-----------------------------------|
| Enter password for omdm user | Пароль для пользователя базы данных (вводимые символы не отображаются) | Произвольный надежный пароль |
| Язык для объектов системы | Язык начальных конфигураций и шаблонов | ru |
| Адрес сервера базы данных | Хост сервера PostgreSQL | localhost (по умолчанию) |
| Порт базы данных | Порт для подключения к PostgreSQL | 5432 (по умолчанию) |
| Пользователь базы данных | Имя пользователя PostgreSQL | omdm |
| Рабочая папка Tomcat 9 | Корневой каталог Tomcat | /var/lib/tomcat9 |
| Папка для скриптов | Каталог для вспомогательных скриптов | /opt/omdm |
| Протокол (http/https) | Протокол доступа к веб-панели | http или https |
| Адрес или DNS-имя веб-панели | IP-адрес или доменное имя сервера в сети | 192.168.1.100 или mdm.example.com |
| Порт веб-панели | Порт, на котором будет работать Tomcat | 8080 (http) или 8443 (https) |
| Путь к порталу | Контекстный путь (ROOT- корневой уровень) | ROOT |
| Настройка SMTP для восстановления | Параметры почтового сервера для отправки уведомлений | n (если не требуется) |

3.3.2. После ввода всех параметров интерактивный мастер предложит подтвердить настройки. При согласии следует нажать «Y» и дождаться завершения установки.

3.3.3. Если выбрано DNS-имя, мастер интерактивный предложит автоматически установить SSL-сертификат через сервис Let's Encrypt. Данная функция возможна только при условии, что сервер доступен из сети Интернет по указанному DNS-имени. Для внутренних сетей следует отказаться от этого шага и установить сертификат вручную.

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

3.4. Проверка работоспособности после установки

3.4.1. После завершения установки проверить работоспособность веб-панели:

```
# Проверка статуса сервиса Tomcat
journalctl -u tomcat9.service

# Открыть в браузере
http://<адрес_сервера>:8080
```

Должна отобразиться страница авторизации СПУ «MDM ОКО». Учетные данные по умолчанию: логин admin, пароль admin. Пароль необходимо сменить немедленно после первого входа.

3.4.2. Сделать резервную копию XML-конфигурации Tomcat:

```
cp /var/lib/tomcat9/conf/Catalina/localhost/ROOT.xml
/var/lib/tomcat9/conf/Catalina/localhost/ROOT.xml~
```

Примечание - В Tomcat 9 существует проблема, при которой XML-конфигурация может быть удалена при обновлении WAR-файла. Резервная копия предотвращает потерю настроек подключения к базе данных.

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

4. УСТАНОВКА СЕРВЕРНОЙ ЧАСТИ С ИСПОЛЬЗОВАНИЕМ DOCKER-КОНТЕЙНЕРА

Данный способ установки предназначен для развертывания СПУ «MDM ОКО» в среде контейнеризации Docker. Контейнер включает все необходимые компоненты (Tomcat, PostgreSQL, tileserver-gl) и управляется с помощью supervisord. Данный способ рекомендуется для быстрого развертывания и переноса системы между средами.

4.1. Установка зависимостей

4.1.1. Для установки зависимостей следует переключиться в режим суперпользователя и установить необходимые пакеты:

```
su
apt-get update
apt-get install -y sudo wget docker docker.io curl jq
```

4.2. Загрузка и распаковка дистрибутива

4.2.1. Для загрузки и распаковки дистрибутива следует перейти в каталог /opt и загрузить архив с носителя, содержащего файлы контейнера СПУ:

```
cd /opt
cp /mnt/USB/oko-mdm-files-mfa.tar.gz ./
cp /mnt/USB/oko-mdm-files-mfa.tar.gz.sha256sum ./
```

4.2.2. Проверить контрольную сумму архива:

```
sha256sum -c ./oko-mdm-files-mfa.tar.gz.sha256sum
```

4.2.3. Ожидаемый результат:

```
oko-mdm-files-mfa.tar.gz: OK
```

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

4.2.4. Распаковать архив:

```
tar -xf ./oko-mdm-files-mfa.tar.gz
cd OKOMDM_FILES
```

4.2.5. Структура дистрибутива после распаковки:

```
OKOMDM_FILES/
├─ change-settings.sh      # мастер настройки конфигурации
├─ change-settings.en.sh  # мастер настройки (английская версия)
├─ run.sh                  # скрипт запуска контейнера
├─ docker-image/
│   └─ oko-aio.tar        # образ контейнера
├─ okomdm-data/
│   ├─ apk/               # APK-файлы агентов (launcher, kiosk и др.)
│   ├─ crt/               # папка для сертификатов TLS
│   ├─ install-config.json # конфигурация установки ОКО МДМ
│   ├─ oko-mdm-server_*.deb # пакет серверной части
│   └─ seed-data/        # начальные данные БД
├─ tileserver-data/      # данные картографического сервера
│   ├─ config.json
│   ├─ fonts/
│   └─ mbtiles/
└─ styles/
└─ version.info          # информация о версии
```

4.3. Настройка конфигурации

4.3.1. Для настройки параметров запуска выполнить скрипт-мастер из каталога OKOMDM_FILES:

```
./change-settings.sh
```

4.3.2. Скрипт последовательно запрашивает параметры, перечисленные в таблице 4.1.

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

Таблица 4 - Параметры мастера настройки контейнера СПУ

| Параметр | Описание | Пример значения |
|------------------------|---|------------------------------------|
| Тип операции | none - без изменений; update - обновление с сохранением баз данных; reinstall - переустановка с очисткой баз данных | none |
| Адрес панели | IP-адрес или DNS-имя сервера | 192.168.0.100 или mdm.example.com |
| Порт панели (Tomcat) | Внутренний порт Tomcat внутри контейнера | 8080 |
| Порт карт (tileserver) | Порт картографического сервера | 9090 |
| Публичный порт панели | Внешний порт в URL (при проксировании может отличаться от внутреннего) | 443 (https) или 8080 (http) |
| Публичный порт карт | Внешний порт tileserver в URL | 9090 |
| Протокол панели | http или https | https |
| TILESERVER_PUBLIC_URL | Полный публичный URL картографического сервера | https://mdm.example.com/tileserver |
| Версия ОКО МДМ | Версия устанавливаемого deb-пакета | 6.0.21 |
| Версия агента | Версия APK-файла агента | 7.0.8 |
| DEBUG_MODE | 0 - PostgreSQL закрыт; 1 - PostgreSQL доступен снаружи (только для отладки) | 0 |
| Ключ MapTiler | Ключ API картографического сервиса (при использовании онлайн-карт) | ключ из cloud.maptiler.com |
| Режим тенант | single - одна организация; multi - мультитенантный режим | single или multi |

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

4.3.3. Пример настройки для HTTP-доступа по IP-адресу:

```
./change-settings.sh
# Ответы на вопросы мастера:
Тип операции: none
Адрес панели: 192.168.0.100
Порт панели (Tomcat): 8080
Публичный порт панели (для URL): 8080
Протокол панели: http
Публичный URL карт: http://192.168.0.100:9090
Режим тенант: single
```

4.3.4. Пример настройки для HTTPS-доступа по DNS-имени:

```
./change-settings.sh
# Ответы на вопросы мастера:
Тип операции: none
Адрес панели: mdm.example.com
Порт панели (Tomcat): 8080
Публичный порт панели (для URL): 443
Протокол панели: https
Публичный URL карт: https://mdm.example.com/tileserver
Режим тенант: single
```

4.3.5. Результатом работы скрипта является обновленный файл `okomdm-data/install-config.json`. Рекомендуется проверить его содержимое перед запуском контейнера.

4.4. Установка сертификатов HTTPS

4.4.1. Для работы по протоколу HTTPS необходимо поместить файлы сертификата в каталог `OKOMDM_FILES/okomdm-data/crt/`:

- `fullchain.pem` - цепочка сертификатов;
- `privkey.pem` - закрытый ключ.

4.4.2. Сертификаты следует получить в уполномоченном удостоверяющем центре и передать в указанный каталог. При использовании

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

сертификатов Let's Encrypt допускается применение вспомогательного скрипта OKOMDM_FILES/okomdm-data/crt/letsencrypt-pem.sh:

```
sudo ./okomdm-data/crt/letsencrypt-pem.sh
```

4.4.3. Скрипт последовательно запрашивает:

- доменное имя (или несколько через запятую);
- адрес электронной почты для регистрации в Let's Encrypt;
- способ проверки домена: standalone (запускает временный HTTP-сервер, порт 80 должен быть свободен) или webroot.

Примечание - Получение сертификата Let's Encrypt возможно только при условии, что сервер доступен из сети Интернет по указанному DNS-имени. Для внутренних закрытых сетей необходимо использовать сертификаты собственного удостоверяющего центра.

4.4.4. Нестандартные имена файлов сертификатов задаются через переменные окружения в install-config.json:

```
"ОКО_TLS_CERT": "имя_файла_fullchain.pem",
"ОКО_TLS_KEY": "имя_файла_privkey.pem"
```

4.5. Запуск контейнера и проверка

4.5.1. Запустить контейнер:

```
./run.sh
```

На запрос о способе загрузки образа указать 1 (локально) и ввести путь:

```
./docker-image/oko-aio.tar
```

4.5.2. Проверить состояние контейнера:

```
docker ps
```

После запуска (через 2-3 минуты) контейнер должен перейти в состояние:

```
STATUS: healthy
```

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

4.5.3. Проверить доступность веб-панели и картографического сервера:

```
curl -s -o /dev/null -w '%{http_code}' http://<адрес>:8080/ # ожидается: 200
curl -s -o /dev/null -w '%{http_code}' http://<адрес>:9090/ # ожидается: 200
```

Примечание - Все параметры настройки (адрес, порты, режим) можно изменить без пересборки образа, для этого достаточно обновить `install-config.json` через `change-settings.sh`, установить `install_type = update` и перезапустить контейнер командой: `docker-compose up -d`.

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

5. НАСТРОЙКА HTTPS ДЛЯ ВЕБ-ПАНЕЛИ В DOCKER-КОНТЕЙНЕРЕ

5.1. Для включения HTTPS в контейнерном окружении необходимо выполнить следующую последовательность действий:

5.1.1. Поместить файлы сертификата в каталог:

```
OKOMDM_FILES/okomdm-data/crt/fullchain.pem
OKOMDM_FILES/okomdm-data/crt/privkey.pem
```

5.1.2. Запустить мастер настройки и указать протокол HTTPS:

```
./change-settings.sh
# Пример ответов:
Адрес панели: your.domain.example
Порт панели (Tomcat): 8080
Публичный порт панели (для URL): 443
Протокол панели: https
```

5.1.3. Запустить контейнер:

```
./run.sh
```

5.2. Если сертификат корректный, DNS настроен правильно и порт 443 открыт на файерволле, веб-панель будет доступна по адресу:

```
https://your.domain.example
```

5.3. При использовании обратного прокси (например, nginx): внутренний порт Tomcat задается параметром ОКО_HTTP_PORT, а публичный порт в URL - параметром ОКО_HTTP_PUBLIC_PORT. Например, Tomcat слушает 8080, а nginx пробрасывает 443:

```
"ОКО_HTTP_PORT": "8080",
"ОКО_HTTP_PUBLIC_PORT": "443",
"ОКО_PROTOCOL": "https"
```

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

6. НАСТРОЙКА ЛОКАЛЬНОГО СЕРВЕРА КАРТОГРАФИИ

СПУ «MDM ОКО» поддерживает отображение местоположения мобильных устройств на карте. При отсутствии доступа к сети Интернет или по соображениям безопасности может быть развернут локальный картографический сервер (tileserver-gl). Сервер запускается в отдельном Docker-контейнере.

6.1. Установка и первоначальный запуск

6.1.1. На целевом сервере выполнить следующие команды:

```
sudo su
cd /opt

# Загрузить архив с данными картографии (или доставить иным способом)
axel -n 20 http://repo-ip-arrd/deb/tileserver-data-last.tar.gz
tar -xf ./tileserver-data-last.tar.gz

# Установить Docker (если не установлен)
sudo apt-get install -y docker docker.io
```

6.1.2. В файлах конфигурации заменить адрес сервера на фактический:

- /opt/tileserver-data/styles/osm-liberty/style.json;
- /opt/tileserver-data/config.json.

Заменить строку <https://okomdm.byterg.ru:9090/> на фактический адрес сервера, например <http://192.168.0.101:9090> или <https://mdm.example.com:9090>.

6.1.3. Загрузить образ контейнера и выполнить тестовый запуск:

```
sudo docker load -i /opt/tileserver-data/docker-image/tileserver-gl-
image.tar

# УКАЗАТЬ ФАКТИЧЕСКИЙ ПОРТ (по умолчанию 9090)
docker run --rm -it -p 9090:8080 -v /opt/tileserver-data:/data
localhost/byterg/tileserver-gl-byterg:v2
```

После проверки выйти из контейнера (Ctrl+C).

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

6.2. Настройка автозапуска в виде системного сервиса

6.2.1. Создать файл systemd-сервиса:

```
nano /etc/systemd/system/tile-server-gl.service
```

Содержимое файла:

```
[Unit]
Description=TileServer-GL Container (Docker)
After=network-online.target docker.service
Wants=network-online.target docker.service
Requires=docker.service

[Service]
Type=simple
# Указать имя и группу пользователя, от имени которого запускается Docker
User=user
Group=user
SupplementaryGroups=docker
Restart=on-failure
RestartSec=5s
ExecStartPre=/usr/bin/docker rm -f tile-server-gl
ExecStart=/usr/bin/docker run \
    --name tile-server-gl \
    --publish 9090:8080 \
    --volume /opt/tileserver-data:/data \
    --entrypoint=/usr/src/app/docker-entrypoint.sh \
    --log-driver=journald \
    localhost/byterg/tileserver-gl-byterg:v2
ExecStop=/usr/bin/docker stop -t 10 tile-server-gl
TimeoutStopSec=30
KillMode=none
TimeoutStartSec=0

[Install]
WantedBy=multi-user.target default.target
```

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

6.2.2. Активировать и запустить сервис:

```
sudo systemctl daemon-reload
sudo systemctl enable tile-server-gl.service
systemctl start tile-server-gl.service

# Проверить статус
systemctl status tile-server-gl.service
```

Примечание - Если используется порт, отличный от 9090, заменить значение --publish N:8080 на фактический номер порта. Тот же порт должен быть указан в настройках веб-панели СПУ.

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

7. НАЧАЛЬНАЯ НАСТРОЙКА СПУ ПОСЛЕ УСТАНОВКИ

7.1. Первый вход в веб-панель

7.1.1. После успешной установки открыть в браузере адрес веб-панели.

Например:

```
http://192.168.0.100:8080  
https://mdm.example.com
```

7.1.2. Ввести учетные данные по умолчанию:

- Логин: admin
- Пароль: admin

7.1.3. После первого входа сменить пароль администратора, выполнив следующие действия: нажать на имя пользователя в правом верхнем углу → «Профиль» → задать новый пароль.

7.1.4. Проверить язык интерфейса в настройках профиля. При необходимости сменить на русский язык.

7.2. Первоначальная настройка конфигурации мобильных устройств

7.2.1. Перейти в раздел «Конфигурации» и создать конфигурацию для управляемых мобильных устройств, выполнив следующие действия:

- нажать «Добавить» или «Клонировать» из стандартной конфигурации;
- задать название конфигурации;
- в разделе «Настройки МДМ» выбрать APK-файл агента (MDM-приложение);
- настроить список разрешенных приложений, параметры безопасности, внешний вид рабочего стола;
- нажать «Сохранить».

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

7.2.2. Проверить готовность конфигурации, выполнив следующие действия: нажать кнопку «QR-код» в столбце действий. Если QR-код отображается - конфигурация готова к применению.

Примечание - Если QR-код не отображается, убедиться, что в конфигурации корректно указано МДМ-приложение (агент) и это приложение загружено в систему (раздел «Приложения»).

7.3. Управление пользователями и ролями

7.3.1. Для создания дополнительных учетных записей перейти в раздел «Настройки» → «Пользователи» и нажать «Добавить».

7.3.2. В СПУ предусмотрены следующие уровни доступа:

- admin - полный доступ ко всем функциям системы;
- user - ограниченный доступ в соответствии с настроенными правами;
- observer - доступ только для просмотра;
- superadmin - суперадминистратор для мультитенантного режима (активируется только при `enable.superadmin = true`).

7.3.3. В мультитенантном режиме (multi-tenant) вход с ролью superadmin должен быть явно разрешен в конфигурации. Для Docker-контейнера: задать `ОКО_ENABLE_SUPERADMIN = 1` в `install-config.json`.

Примечание - Для безопасности рекомендуется не активировать superadmin в single-tenant окружениях и сменить стандартные учетные данные после установки.

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

8. УСТАНОВКА АГЕНТА СПУ НА МОБИЛЬНОЕ УСТРОЙСТВО

Агент СПУ «MDM ОКО» (Android APK) устанавливается непосредственно на управляемое мобильное устройство. После установки и регистрации мобильное устройство становится доступным для управления через веб-панель. Поддерживаются два основных способа установки.

8.1. Ручная установка APK

8.1.1. Ручная установка применяется при отсутствии возможности использовать QR-код (например, на Android 4.0–6.x с ограниченными возможностями). Порядок действий:

- подключить мобильное устройство к персональному компьютеру с помощью USB-кабеля;
- на мобильном устройстве включить режим «Передача файлов» (вместо режима зарядки);
- скачать APK-файл агента с веб-панели СПУ (раздел «Приложения», кнопка «Скачать APK»);
- скопировать APK-файл в папку «Загрузки» (Downloads) на мобильном устройстве;
- на мобильном устройстве открыть приложение «Файлы» и запустить скачанный APK-файл;
- нажать «Настройки» → переключить флаг «Разрешить установку из этого источника» → нажать «Установить»;
- после установки нажать «Открыть» → «Продолжить»;
- выбрать «Активировать это приложение администратора» → «Продолжить»;
- активировать разрешение «Управление всеми файлами» → перейти назад;

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

- предоставить разрешения: «Показывать уведомления», «Поверх других приложений», «Будильник и напоминания», «Изменить системные настройки», «Местоположение - при любом режиме», «Камера»;

- в настройках агента задать «Главное приложение - MDM-агент»;

- нажать кнопку «QR-код» и выполнить шаги пп. 8.2.3, 8.2.4.

8.2. Установка через QR-код (рекомендуемый способ)

8.2.1. Установка через QR-код является рекомендуемым способом для мобильных устройств на Android 7.0 и выше. Этот метод обеспечивает полный функционал управления (режим «Device Owner») без ручной настройки прав.

8.2.2. Перед установкой через QR-код следует подготовить мобильное устройство, выполнив следующие действия:

- убедиться, что мобильное устройство заряжено не менее чем на 20 процентов;

- выполнить сброс к заводским настройкам (если мобильное устройство ранее использовалось) или использовать новое мобильное устройство на экране приветствия;

- на экране приветствия нажать 7 раз по тексту «Привет» (или аналогичному элементу) - откроется скрытый интерфейс установки администратора через QR-код.

8.2.3. В веб-панели СПУ создать или выбрать мобильное устройство, выполнив следующие действия:

- перейти в раздел «Устройства»;

- нажать «Добавить» - ввести уникальный идентификатор мобильного устройства (например, device001);

- выбрать конфигурацию (например, «Управляемый лаунчер»);

- нажать «Сохранить»;

- нажать кнопку «QR-код» в столбце действий для созданного мобильного устройства.

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

8.2.4. На управляемом мобильном устройстве навести камеру на отображенный QR-код:

- при запросе настройки Wi-Fi - подключить мобильное устройство к сети;
- произойдет автоматическая загрузка и установка плагинов агента;
- на каждый запрос установки плагина нажать «Установить», затем «Готово»;
- по завершении установки на мобильном устройстве отобразится рабочий стол, настроенный в выбранной конфигурации.

8.3. Настройка прав администратора мобильного устройства (Device Owner)

8.3.1. В некоторых случаях (например, при использовании нестандартных сборок Android) права Device Owner не устанавливаются автоматически и требуют ручного назначения через ADB.

8.3.2. Включить отладку по USB на мобильном устройстве:

- открыть «Настройки» → «О телефоне»;
- нажать 10 раз по пункту «Номер сборки» до появления сообщения «Вы стали разработчиком»;
- вернуться в «Настройки» → «Система» → «Для разработчиков»;
- активировать пункт «Отладка по USB» → подтвердить.

8.3.3. Подключить мобильное устройство к персональному компьютеру с помощью USB-кабеля и выдать права Device Owner через ADB:

```
# Убедиться, что устройство определяется
adb.exe devices

# Выдать права Device Owner агенту OKO МДМ
adb.exe shell dpm set-device-owner com.hmdm.launcher/.AdminReceiver
```

Примечание - Права Device Owner нельзя назначить, если на мобильном устройстве уже добавлены пользовательские аккаунты (Google и другие). Перед назначением прав необходимо удалить все аккаунты из настроек.

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

9. ДОПОЛНИТЕЛЬНЫЕ ОПЕРАЦИИ

9.1. Импорт и экспорт конфигураций

9.1.1. СПУ поддерживает сохранение конфигураций в формате JSON для переноса между серверами или резервного копирования.

9.1.1.1. Для экспорта конфигурации выполнить следующие действия:

- перейти на вкладку «Конфигурации»;
- в столбце «Действия» нажать кнопку «Экспорт»;
- в диалоговом окне нажать «Сохранить».

9.1.1.2. Для импорта конфигурации выполнить следующие действия:

- перейти на вкладку «Конфигурации»;
- нажать кнопку «Импорт»;
- выбрать JSON-файл конфигурации;
- выбрать действие: «Создать новую конфигурацию» или «Обновить существующую» (с выбором целевой конфигурации из списка);
- при необходимости изменить имя конфигурации;
- нажать «Импорт».

9.1.2. После импорта проверить готовность конфигурации нажатием кнопки «QR-код» в строке импортированной конфигурации.

Примечание - JSON-файл конфигурации не содержит APK-файлов приложений и бинарных ресурсов. Если на целевом сервере отсутствуют приложения, указанные в конфигурации, QR-код не будет отображен. Необходимо добавить приложения вручную через раздел «Приложения» и назначить их конфигурации.

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

9.2. Добавление файлов обновлений программного обеспечения мобильных устройств

9.2.1. Для управления обновлениями программного обеспечения мобильных устройств ОКО (ОТА-обновления) необходимо выполнить следующие действия:

9.2.2. Подготовить файлы программного обеспечения:

```
# Пример: создать файл контрольной суммы для архива прошивки
f=./ОКО-SE-13.999.999.5.oko_tab_5_pro.ota.zip
sha256sum -- "$f" | awk '{print $1}' > "${f}.sha256"

# Проверить содержимое файла контрольной суммы
cat "${f}.sha256"
```

9.2.3. Загрузить файлы программного обеспечения в веб-панель СПУ, выполнив следующие действия:

- перейти на вкладку «Файлы»;
- нажать «Добавить»;
- активировать галку «Загрузка файлов прошивок»;
- указать тип мобильного устройства (например, ОКО Tab 5 PRO);
- нажать «Файл» → выбрать файл ZIP и файл SHA256 одновременно (через <Ctrl> для множественного выбора);
- убедиться в появлении сообщения «Контрольная сумма совпадает»;
- нажать «Скопировать на сервер» → дождаться завершения загрузки;
- нажать «Сохранить» → дождаться формирования файла builds.json.

9.2.4. Назначить образ конфигурации, выполнив следующие действия:

- перейти в конфигурацию соответствующего типа мобильного устройства;
- скопировать строку-путь к файлу builds.json из поля «Другие записи QR-кода»;

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

- проверить доступность установочного файла программного обеспечения по прямой ссылке (открыть URL в браузере или выполнить wget).

9.3. Регистрация ключа картографического сервиса MapTiler

9.3.1. При использовании онлайн-карт MapTiler (вместо локального тайл-сервера) необходимо зарегистрировать API-ключ, выполнив следующие действия:

- перейти на портал <https://cloud.maptiler.com>;
- зарегистрироваться или войти в существующую учетную запись;
- перейти в раздел <https://cloud.maptiler.com/account/keys/>;
- нажать «New key» → создать ключ с подходящим именем;
- скопировать полученный ключ.

9.3.2. Ввести ключ в настройки СПУ:

- при установке из deb-пакета указать при запросе мастера установки или задать в файле конфигурации Tomcat;
- при установке через Docker-контейнер указать параметр ОКО_TILER_KEY в install-config.json при запуске скрипта change-settings.sh.

9.4. Обновление серверной части СПУ

9.4.1. Для обновления СПУ, установленной из deb-пакета, выполнить следующие действия:

```
# При наличии репозитория
sudo apt-get update && sudo apt-get upgrade oko-mdm-server

# При наличии нового deb-пакета
sudo apt-get install ./oko-mdm-server_X.Y.Z_amd64.deb

# Перезапустить Tomcat
sudo systemctl restart tomcat9
```

Примечание - Перед обновлением создать резервную копию конфигурации Tomcat: `cp /var/lib/tomcat9/conf/Catalina/localhost/ROOT.xml ~/ROOT.xml.bak`

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

9.4.2. Для обновления СПУ в Docker-контейнере, выполнить следующие действия:

- скопировать новый deb-пакет в каталог OKOMDM_FILES/okomdm-data/;
- в install-config.json задать: "install_type": "update" и обновить поле "deb" на имя нового пакета;
- перезапустить контейнер командой: docker-compose up -d;
- дождаться сообщения [init] Initialization complete в журнале контейнера.

Примечание - Тип операции "update" сохраняет данные базы данных. Для полной переустановки с очисткой базы данных использовать "install_type": "reinstall".

| | | | | |
|------|------|----------|-------|------|
| | | | | |
| | | | | |
| Изм. | Лист | № докум. | Подп. | Дата |

