

Инструкция по инсталляции

Установка должна производиться на операционную систему Linux Ubuntu 16.04 (64 bit).

Возможны два варианта установки приложения: на отдельный сервер, в комплексе с другими приложениями на один сервер.

1. Установка на отдельный сервер

Перед установкой убедитесь. Что в системе установлены следующие пакеты:

- **libpq5**
- **openjdk-7-jre**
- **mongodb**
- **Postgresql-9.6**
- **Postgis-2.1**
- **Nginx**

Описание установки пакетов:

- <https://www.postgresql.org/download/linux/debian/>
- <http://postgis.net/install/>

Остальные пакеты устанавливаются из стандартных репозиториев.

Процедура установки:

1. Установка управляющего демона: **dpkg -i daemon_0.6.4-1_amd64.deb**
2. Установка коллектора навигационных и телеметрических данных: **dpkg -i automapd2_2.4.wheezy147_amd64.deb**
3. Установка Infotech.Auto.engine : **dpkg -i --ignore-depends=mongodb-10gen automap-web_2.4.5-FINAL-49_all.deb**

Установка RabbitMQ:

- `echo "deb http://www.rabbitmq.com/debian/ testing main" >> /etc/apt/sources.list`
- `curl http://www.rabbitmq.com/rabbitmq-signing-key-public.asc | sudo apt-key add -`
- `wget https://dl.bintray.com/rabbitmq/Keys/rabbitmq-release-signing-key.asc | sudo apt-key add -`
`apt-get update`
`sudo apt-get install rabbitmq-server`

- Включить Веб-Админ Rabbit#
`sudo rabbitmq-plugins enable rabbitmq_management`
[http://\[IP_адрес_сервера\]:15672/](http://[IP_адрес_сервера]:15672/) - по-этому адресу доступ

- Стандартные имя и пароль – guest

1. Запуск:

```
service rabbitmq-server start
```

2. Остановка:

```
service rabbitmq-server stop
```

3. Перезапуск:

```
service rabbitmq-server restart
```

4. Проверка статуса:

```
service rabbitmq-server status
```

```
#Добавить пользователя#
```

```
rabbitmqctl add_user login password
```

```
rabbitmqctl set_user_tags login administrator
```

```
rabbitmqctl set_permissions -p / login "." "." ".*"
```

Настройка конфигурационных файлов:

1. Необходимо внести изменения в файл `/etc/automapd2/config.xml`, указать учетные данные для подключения к MongoDB и PostgreSQL (host, port, dbname, username, password)
2. Необходимо внести изменения в файл `/etc/automap-web/application.conf`, указать учетные данные для подключения к PostgreSQL (db.default.url, db.default.user, db.default.password)
3. Добавить в конфиг web сервера Nginx следующую конфигурацию:

```
server {
    error_log /var/log/nginx/automap.error.log info;
    access_log /var/log/nginx/automap.error.log;
    listen 80;
    server_name server.domain.name;
    location / {
        proxy_http_version 1.1;
        proxy_set_header X-Real-IP $remote_addr;
```

```
proxy_set_header Host $host;
proxy_set_header Upgrade $http_upgrade;
proxy_set_header Connection "upgrade";
proxy_pass http://127.0.0.1:9000/;
}
}
```

2. Установка в комплексе с другими приложения на один сервер

Перед непосредственной установкой должны быть установлены остальные приложения комплекса и следующие пакеты:

- **libpq5**
- **openjdk-7-jre**
- **mongodb**

Процедура установки:

1. Установка управляющего демона: **dpkg -i daemon_0.6.4-1_amd64.deb**
2. Установка коллектора навигационных и телеметрических данных: **dpkg -i automapd2_2.4.wheezy147_amd64.deb**
3. Установка Infotech.Auto.engine : **dpkg -i --ignore-depends=mongodb-10gen automap-web_2.4.5-FINAL-49_all.deb**

Настройка конфигурационных файлов:

1. Необходимо внести изменения в файл **/etc/automapd2/config.xml**, указать учетные данные для подключения к Mongodb и Postgresql (host, port, dbname, username, password)
2. Необходимо внести изменения в файл **/etc/automap-web/application.conf**, указать учетные данные для подключения к Postgresql (db.default.url, db.default.user, db.default.password)

Перед правкой конфигурации веб сервера nginx необходимо узнать IP адрес сервера в сети докеров, для этого надо выполнить команду **ifconfig -a**. В выводе команды надо найти интерфейс с названием **docker0**, значением параметра **inet addr** которого является искомый IP адрес.

Теперь необходимо создать файл `/usr/local/infotech/localdata/etc/nginx/conf.d/automap.conf` со следующим содержимым:

```
server {
    listen 80;
    server_name server.domain.name;
    location / {
        proxy_http_version 1.1;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header Host $host;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection "upgrade";
        proxy_pass http://IP адрес сервера:9000/;
    }
}
```

Перезапустить nginx: **docker restart nginx**

Для обоих типов установки приложения дальнейшая настройка совпадает.

Запуск приложений:

1. Запустить engine : `/etc/init.d/automap-web start`
2. Запустить коллектор : `/etc/init.d/automapd start`

3. Исходный файл для развёртывания Мульти адаптера.

Программа считывает данные из исходного файла, а потом заносит их в базу данных. При следующем запуске программы, настройки читаются из базы данных.

```
=====
; Раздел [Adapter] - Основные параметры программы
=====;
Sensors - Количество датчиков - Целое число
```

```

; dbActive - Активизация баз данных - Main - основная база данных, Backup -
резервная база данных, Both - обе базы данных
; Processing - Обработка данных - Bit[0] - объекты, Bit[1] - инциденты,
Bit[2] - статистика, Bit[3] - PVR
; Typing - Вывод данных на экран - Bit[0] - объекты, Bit[1] - инциденты,
Bit[2] - статистика, Bit[3] - PVR, Bit[4] - данные, Bit[5] - отладочная
информация
; Printing - Вывод данных в log файл - Bit[0] - объекты, Bit[1] - инциденты,
Bit[2] - статистика, Bit[3] - PVR, Bit[4] - данные, Bit[5] - отладочная
информация
; Sending - Вывод данных в базу - Bit[0] - объекты, Bit[1] - инциденты,
Bit[2] - статистика, Bit[3] - PVR
; Filter - Флаг фильтрации статистики - Yes - данные фильтруются, No -
выводятся все данные
;=====
[Adapter]
Sensors = 1
dbActive = None
Processing = 15
Typing = 15
Printing = 15
Sending = 0
Filter = Yes

;=====
; Раздел [MainDB] - Параметры основной базы данных
;=====
; Server - IP адрес сервера базы данных - Текстовая строка
; Port - Порт сервера базы данных - Целое число
; Name - Имя базы данных - Текстовая строка
; User - Имя пользователя базы данных - Текстовая строка
; Password - Пароль на базу данных - Текстовая строка
;=====
[MainDB]
Server = localhost
Port = 5432
Name = smartmicro
User = smartmicro
Password = smartmicro

```

```

;=====
; Раздел [BackupDB] - Параметры резервной базы данных
;=====
; Server - IP адрес сервера базы данных - Текстовая строка
; Port - Порт сервера базы данных - Целое число
; Name - Имя базы данных - Текстовая строка
; User - Имя пользователя базы данных - Текстовая строка
; Password - Пароль на базу данных - Текстовая строка
;=====

[BackupDB]
Server = 172.242.40.106
Port = 5432
Name = smartmicro
User = smartmicro
Password = smartmicro

;=====;
Раздел [Sensor] - Основные параметры датчика
;=====;
ID - ID датчика - Целое число
; Name - Название датчика - Текстовая строка
; Type - Тип датчика - UMR-0A, UMR-0C, UMR-0F
; Interface - Интерфейс подключения датчика - Ethernet, Serial
; Extended - Расширенный набор данных об объекте - Yes - дополнительные
данные, No - стандартный набор данных
;=====

[Sensor]
ID=10
Name = Kazakhstan
Type = UMR-0C
Interface = Ethernet
Extended = No

;=====;
Раздел [Ethernet] - Параметры Ethernet соединения
;=====;
Address - IP адрес датчика - Текстовая строка
; Port - Порт датчика - Целое число

```

```

; IsServer - Режим подключения датчика - Yes - датчик в режиме сервера, No -
датчик в режиме клиента
; Timeout - Задержка чтения данных [мсек] - Целое число
; Delay - Задержка подключения датчика [мсек] - Целое число
;=====
[Ethernet]
Address = 10.134.42.151
Port = 55555
IsServer = No
Timeout = 100
Delay 300000

;=====;
Раздел [Serial] - Параметры соединения по COM порту
;=====;
Port - Порт - Текстовая строка
; BaudRate - Скорость передачи данных - Целое число
; DataBits - Размер посылки - Целое число
; StopBits - Число стоповых битов - Целое число
; Parity - Проверка чётности - Yes, No
; Duplex - Режим работы порта - Yes - Full Duplex, No - Half Duplex
; Timeout - Задержка чтения данных [мсек] - Целое число
; Delay - Задержка пере-подключения датчика [мсек] - Целое число
;=====
[Serial]
Port = COM5
BaudRate = 115200
DataBits = 8
StopBits = 0
Parity = No
Duplex = Yes
Timeout = 50
Delay 1000

;=====;
; Раздел [Speed] - Параметры предельных скоростей
;=====;
Max - Максимально допустимая скорость [км/ч] - Натуральное число
; Min - Минимально допустимая скорость [км/ч] - Натуральное число

```

```
;=====
[Speed]
Max=100
Min=60
```

4. Развёртывание базы данных

Скрипт для создания базы данных:

```
CREATE DATABASE smartmicro
WITH
OWNER = postgres
ENCODING = 'UTF8'
LC_COLLATE = 'en_US.UTF-8'
'Russian_Russia.1251'
LC_CTYPE = 'en_US.UTF-8'
TABLESPACE = pg_default
CONNECTION LIMIT = -1;
GRANT TEMPORARY, CONNECT ON DATABASE smartmicro TO PUBLIC;
GRANT ALL ON DATABASE smartmicro TO postgres;
```

Создания дополнительных пользователей:

```
CREATE USER smartmicro WITH
LOGIN
SUPERUSER
INHERIT
CREATEDB
CREATEROLE
REPLICATION
VALID UNTIL 'infinity'
```