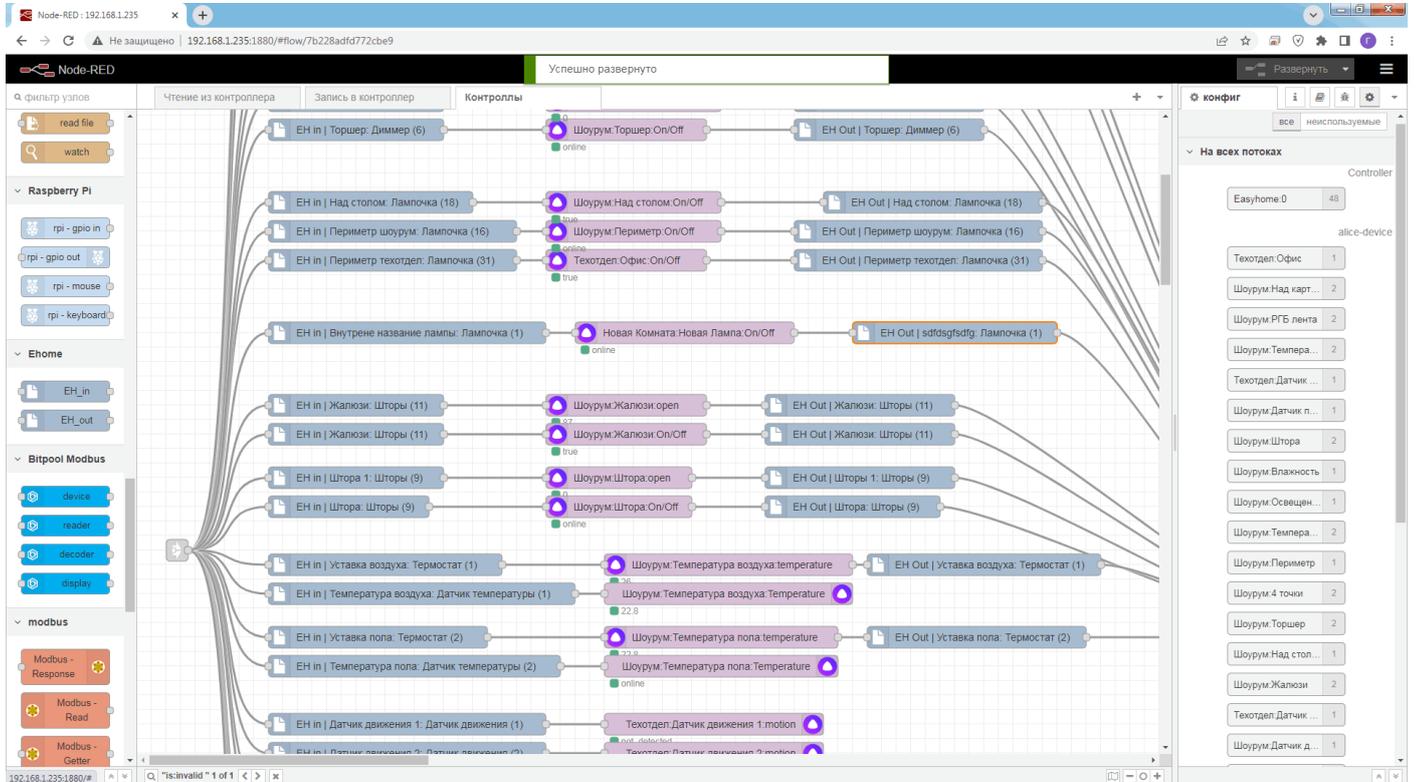


# Настройка шлюза связи EasyHomeCS на базе [Node-RED](#) для связи умного дома [EasyHome](#) с [Яндекс-Дом](#) и [голосовым помощником Алиса](#)



Санкт-Петербург, Россия, 2025



Данный шлюз обеспечивает постоянную связь с контроллером системы EasyHome через интерфейс Ethernet по протоколу ModbusTCP и обеспечивает взаимодействие с сервисом Яндекс-ДОМ и голосовым помощником АЛИСА.

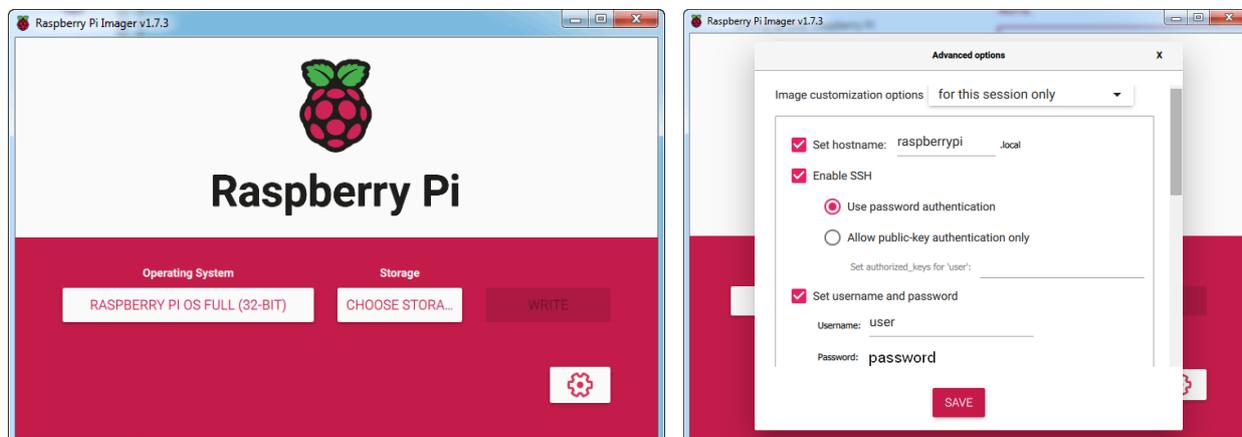
## СОДЕРЖАНИЕ

1.	Подготовка оборудования шлюза Node-RED на базе Raspberry.....	3
2.	Потоки чтения и записи в контроллер.....	10
3.	Объекты связи Node-RED с контроллером .....	11
4.	Вэб-визуализация Яндекс-ДОМ .....	18
5.	Визуализация в приложении Яндекс-ДОМ .....	18
6.	Настройка PUSH-уведомлений .....	20
6.1.	Настройка PUSH уведомлений .....	20
6.2.	Привязка пользовательских устройств .....	21
6.3.	Настройка времени в Raspberry Pi.....	22
6.4.	Настройка сохранения статуса PUSH уведомлений.....	23

**Если Вы заказали готовый мини-сервер EasyHomeCS, то можете пропустить пункт 1.**

## 1. Подготовка оборудования шлюза Node-RED на базе Raspberry.

1. Установите **Raspberry PI OS FULL (32-BIT)** на **Raspberry** используя установщик образа на microSD с официального сайта <https://www.raspberrypi.com/software/>, при установке включите опцию **hostname** и **SSH** (терминал удалённого доступа), пользователь **user**, пароль **password**.



! Поищите в меню установщика, оно периодически обновляется, если в установщике образов Raspberry есть вариант **Raspberry PI OS** с предустановленным **NodeRED** – используйте его, вводя все данные по авторизации при инициализации образа.

! ПО шлюза Node-RED может быть установлено так же на ПК с Windows и другие ОС – см информацию на сайте [Node-RED local](https://nodered.local).

! Для питания Raspberry Pi 3 B+ используйте рекомендованный БП 5В на 3А (хотя средний потребляемый ток обычно 1.5А)

## 2. Настройка IP адреса для Raspberry

На устройстве Raspberry Pi на сетевом интерфейсе Ethernet стоит по-умолчанию получение IP адреса автоматически от сервиса DHCP (от роутера сети). Узнайте, какой IP адрес присвоен устройству, это можно сделать несколькими способами:

- 2.1. Подключите **монитор FullHD через HDMI** гнездо и клавиатуру с мышкой, Вы увидите рабочий стол системы RaspberryOS, задайте в настройках фиксированный IP адрес устройству из локальной подсети (рекомендуем **192.168.1.235**) и перезагрузите устройство для активации изменений.
- 2.2. При отсутствии монитора узнать **временный IP** адрес можно через роутер в списке клиентов или сканером сети. Для смены IP адреса на фиксированный необходимо выполнить ряд действий:

- 2.2.1. Подключиться по **временному IP** адресу терминалом **PuTTY** по **SSH** или файловым менеджером **WinSCP** по **SFTP**, пользователь **user**, пароль **password**.  
 (ссылка на **PuTTY** - <https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>)  
 (ссылка на **WinSCP** - <https://winscp.net/eng/download.php>)

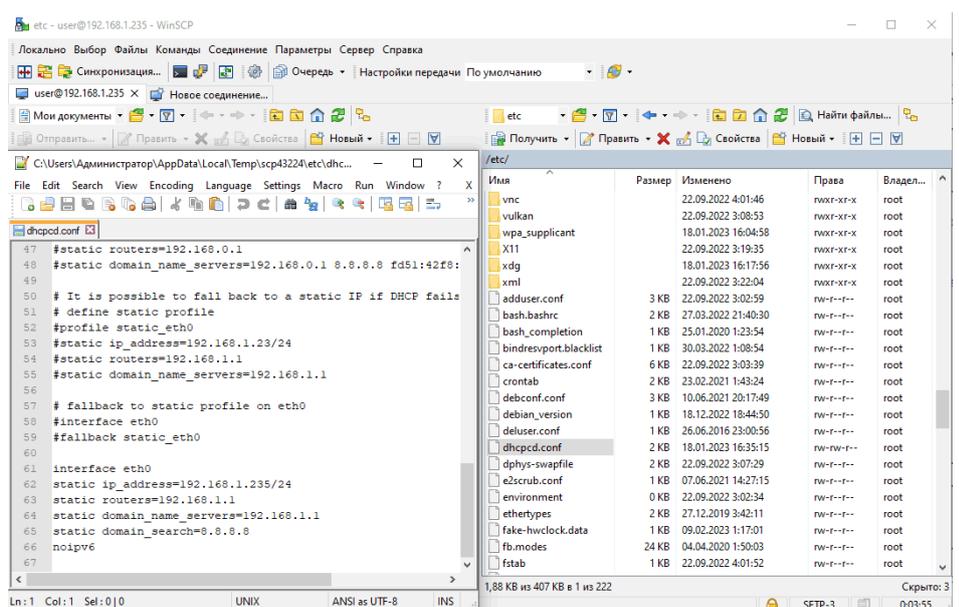
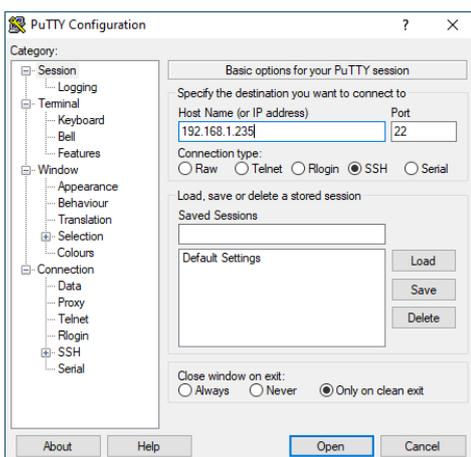
2.2.2. Открыть файл с настройками сети в редакторе командой:  
**sudo nano /etc/dhcpd.conf**, ввести пароль **password**, если потребуется.

2.2.3. В открывшемся редакторе дописать в файл или заменить настройки:  
 interface eth0  
 static ip\_address=**192.168.1.235/24**  
 static routers=192.168.1.1  
 static domain\_name\_servers=192.168.1.1  
 static domain\_search=8.8.8.8  
 noipv6

! пролистать файл до конца, чтоб убедиться в отсутствии старых строк !

2.2.4. Сохранить изменения (**ctrl + O**), закрыть файл (**ctrl + X**).

2.2.5. Перезагрузить raspberry командой **sudo reboot** или питанием.

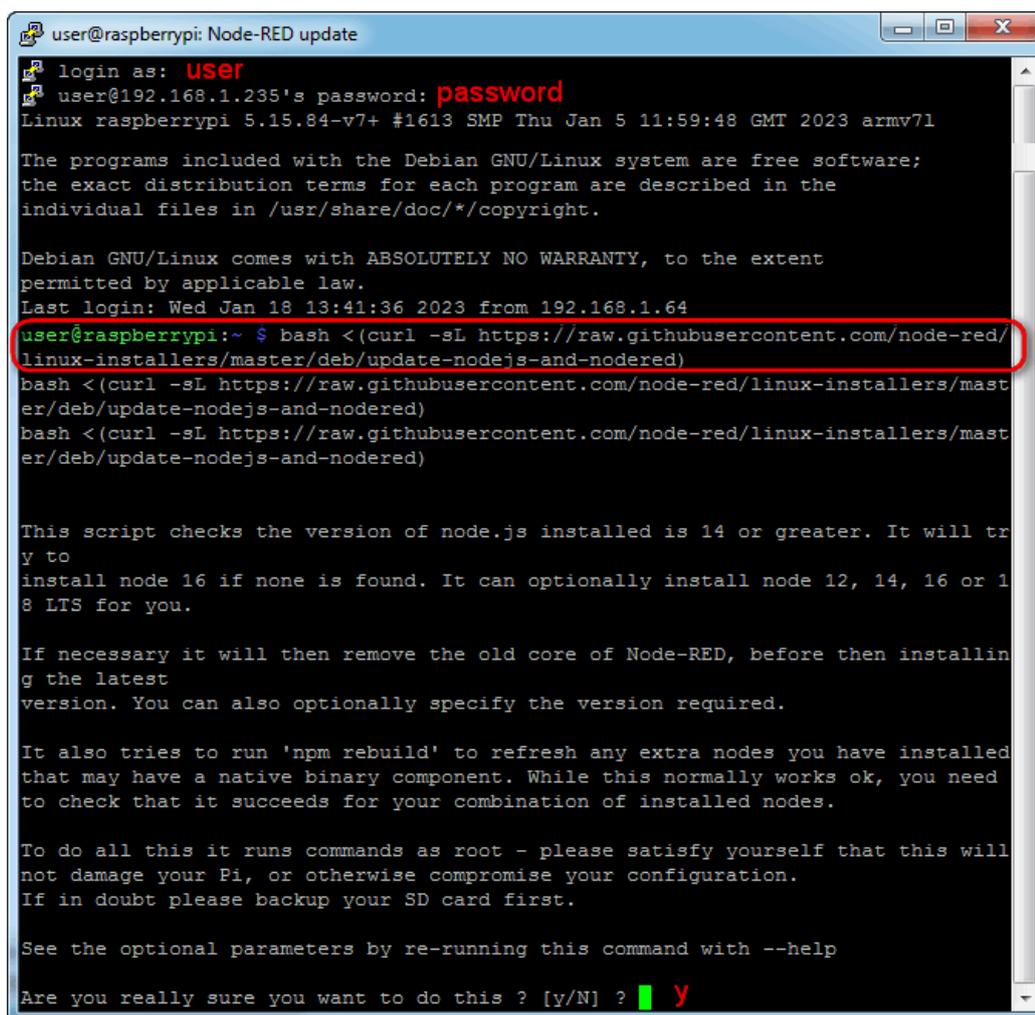


### 3. Установка Node-RED на Raspberry

- 3.1. Подключитесь по SSH к Raspberry (используйте терминал PuTTY, локальный IP 192.168.1.235, пользователь **user**, пароль **password**)
- 3.2. Запустите команду для скрипта скачивания и установки nodejs-and-nodered. Подробное описание установки есть на сайте <https://nodered.org/docs/getting-started/raspberrypi>. Команда на 08.12.2024 выглядит так :

[bash <\(curl -sL https://raw.githubusercontent.com/node-red/linux-installers/master/deb/update-nodejs-and-nodered\) --node20](https://raw.githubusercontent.com/node-red/linux-installers/master/deb/update-nodejs-and-nodered)

Вводим, подтверждаем:



```
user@raspberrypi: Node-RED update
login as: user
user@192.168.1.235's password: password
Linux raspberrypi 5.15.84-v7+ #1613 SMP Thu Jan 5 11:59:48 GMT 2023 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Jan 18 13:41:36 2023 from 192.168.1.64
user@raspberrypi:~ $ bash <(curl -sL https://raw.githubusercontent.com/node-red/
linux-installers/master/deb/update-nodejs-and-nodered)
bash <(curl -sL https://raw.githubusercontent.com/node-red/linux-installers/mast
er/deb/update-nodejs-and-nodered)
bash <(curl -sL https://raw.githubusercontent.com/node-red/linux-installers/mast
er/deb/update-nodejs-and-nodered)

This script checks the version of node.js installed is 14 or greater. It will tr
y to
install node 16 if none is found. It can optionally install node 12, 14, 16 or 1
8 LTS for you.

If necessary it will then remove the old core of Node-RED, before then installin
g the latest
version. You can also optionally specify the version required.

It also tries to run 'npm rebuild' to refresh any extra nodes you have installed
that may have a native binary component. While this normally works ok, you need
to check that it succeeds for your combination of installed nodes.

To do all this it runs commands as root - please satisfy yourself that this will
not damage your Pi, or otherwise compromise your configuration.
If in doubt please backup your SD card first.

See the optional parameters by re-running this command with --help
Are you really sure you want to do this ? [y/N] ? y
```

- 3.3. После установки отказываемся менять настройки по-умолчанию и ещё раз подтверждаем установку, отказываемся от специфических опций. Устанавливаем АВТОЗАПУСК Node-RED при загрузке командой **sudo systemctl enable nodered.service**. Перезагружаем Raspberry командой **sudo reboot**.

```
user@raspberrypi: Node-RED update

Running Node-RED install for user user at /home/user on raspbian
Stop Node-RED                               ✓
Remove old version of Node-RED              ✓
Remove old version of Node.js               ✓
Install Node.js 16 LTS                       ✓   v16.19.0   Npm 8.19.3
Clean npm cache                             ✓
Install Node-RED core                       ✓   3.0.2
Move global nodes to local                  -
Npm rebuild existing nodes                  ✓
Install extra Pi nodes                      ✓
Add shortcut commands                       ✓
Update systemd script                       ✓

Any errors will be logged to /var/log/nodered-install.log
All done.
You can now start Node-RED with the command node-red-start
or using the icon under Menu / Programming / Node-RED
Then point your browser to localhost:1880 or http://{your_pi_ip-address}:1880
*****
### WARNING ###
DO NOT EXPOSE NODE-RED TO THE OPEN INTERNET WITHOUT SECURING IT FIRST

Even if your Node-RED doesn't have anything valuable, (automated) attacks will
happen and could provide a foothold in your local network
- After running Node-RED for the first time, change the ownership of the settings
file to 'root' to prevent unauthorised changes:

    sudo chown root:root ~/.node-red/settings.js

*****
Would you like to customise the settings now (y/N) ? n
Settings not initialized.

This script checks the version of node.js installed is 14 or greater. It will try to
install node 16 if none is found. It can optionally install node 12, 14, 16 or 18
LTS for you.

To do all this it runs commands as root - please satisfy yourself that this will
not damage your Pi, or otherwise compromise your configuration.
If in doubt please backup your SD card first.

See the optional parameters by re-running this command with --help

Are you really sure you want to do this ? [y/N] ? y

Would you like to install the Pi-specific nodes ? [y/N] ? n

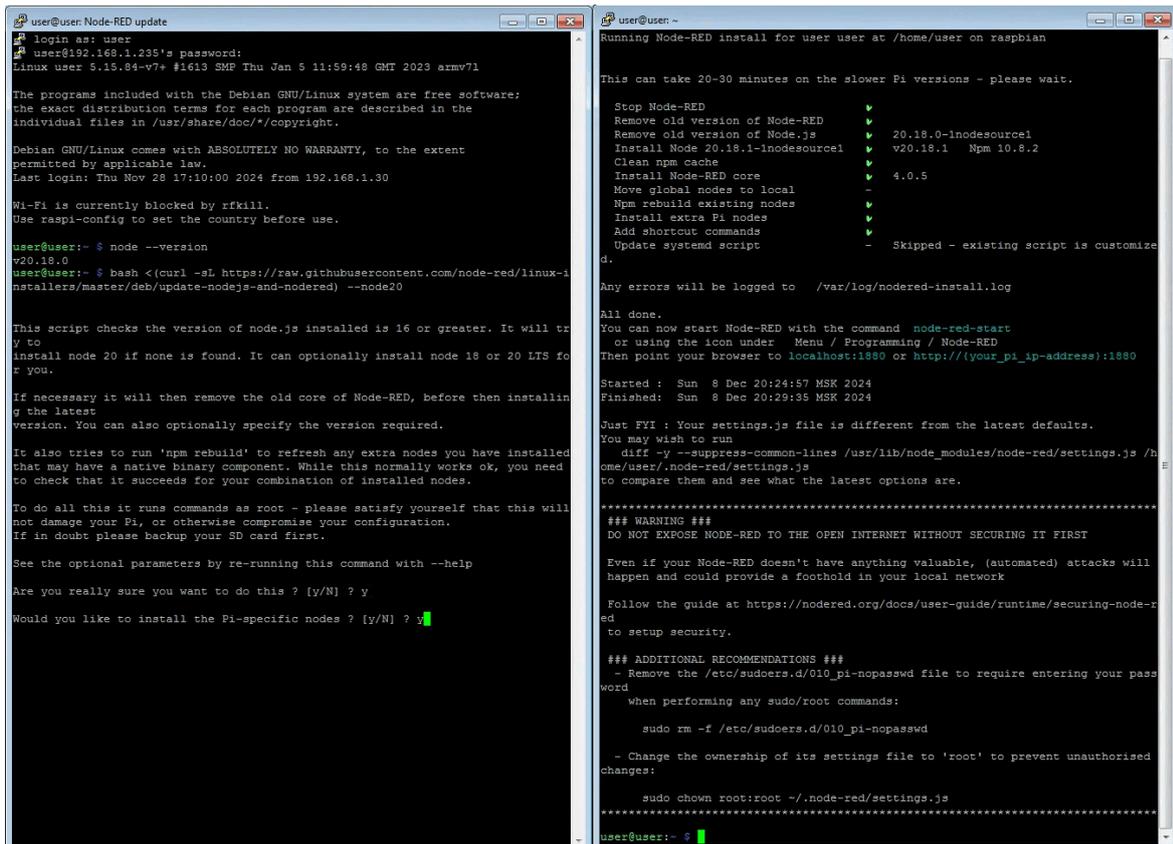
*****
user@raspberrypi:~$ sudo systemctl enable nodered.service
Created symlink /etc/systemd/system/multi-user.target.wants/nodered.service → /lib/systemd/system/nodered.service.
user@raspberrypi:~$ sudo reboot
```

#### 4. Обновление Node-RED 3.x до NodeRED 4.x с nodejs 20

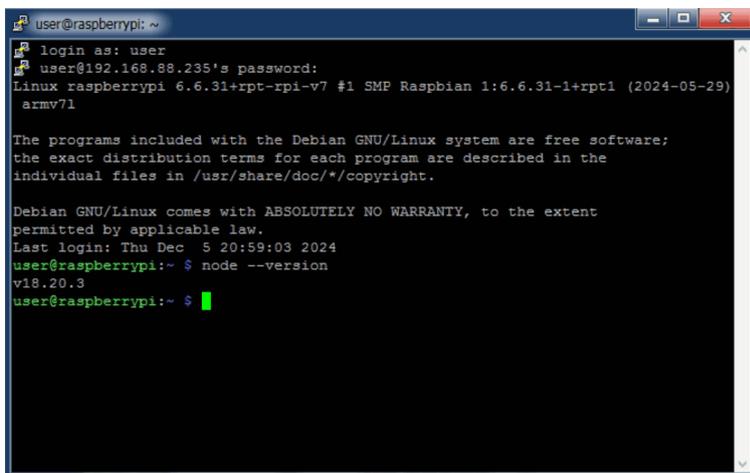
В ряде случаев (для работы PUSH) может потребоваться обновить уже установленный пакет Node-RED 3 до версии Node-RED 4 и модуля nodejs до 20й версии. Это выполняется командой через доступ по SSH:

```
bash <(curl -sL https://raw.githubusercontent.com/node-red/linux-installers/master/deb/update-nodejs-and-nodered) --node20
```

(Скопировать ссылку в буфер обмена и в SSH нажать правую кнопку мышки для вставки)

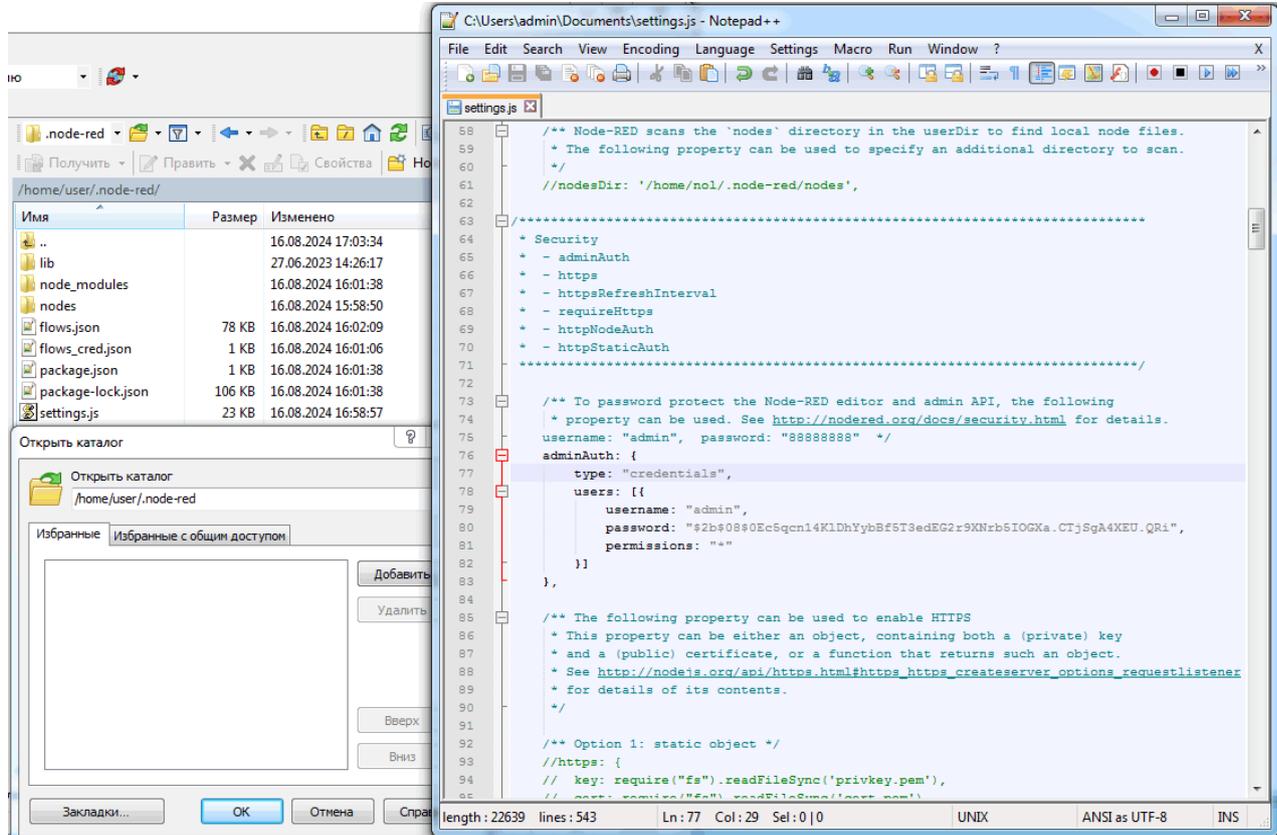


#### 5. Проверка версии NodeJS в Node-RED



## 6. Установка пароля на Web-интерфейс

Для установки пароля необходимо открыть **скрытый** каталог `/home/user/.node-red` и в файле `settings.js` в разделе Security (строки 63..90) раскомментировать скрипт и указать имя пользователя и хэш-код пароля, по умолчанию устанавливаем **admin 88888888**, для этого пароля указываем хэш-код **\$2b\$08\$0Ec5qcn14KIDhYybBf5T3edEG2r9XNrb5IOGKa.CTjSgA4XEU.QRi** :

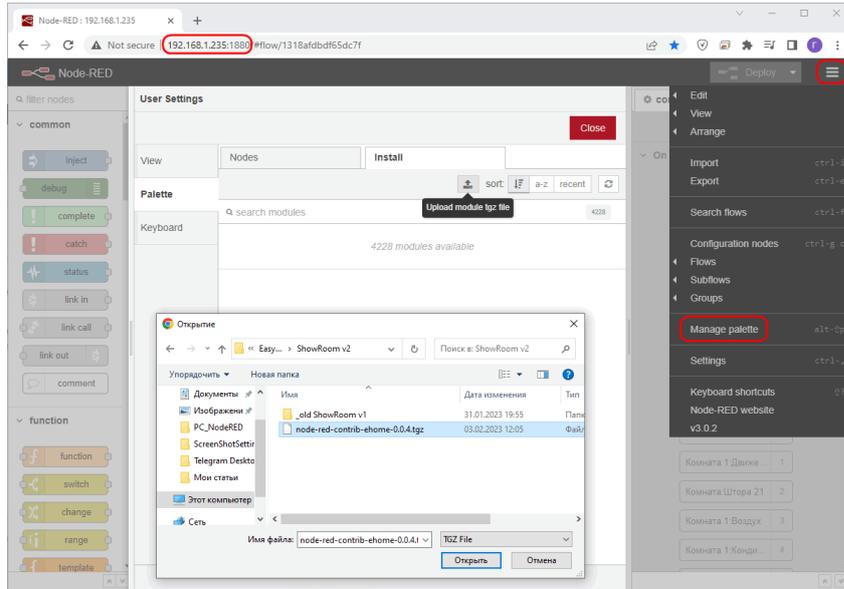


## 7. Запуск Node-RED и установка пакета node-red-contrib-ehome

- 7.1. В браузере на ПК или на Raspberry открываем интерфейс Node-RED указывая в строке адреса <http://192.168.1.235:1880/> (Пароль по умолчанию **admin 88888888**)
- 7.2. Устанавливаем пакет драйвера и примеров **node-red-contrib-ehome-0.0.24.tgz** (или более новый) через меню: **MENU -> Manage palette -> Palette -> Install -> Upload module tgz file**

Проверяем в этом же окне **User Settings** (Параметры пользователя) в списке **Nodes** (Узлы) наличие **node-red-contrib-ehome**

**! После обновления пакета необходимо полностью перезагрузить Node-RED !**

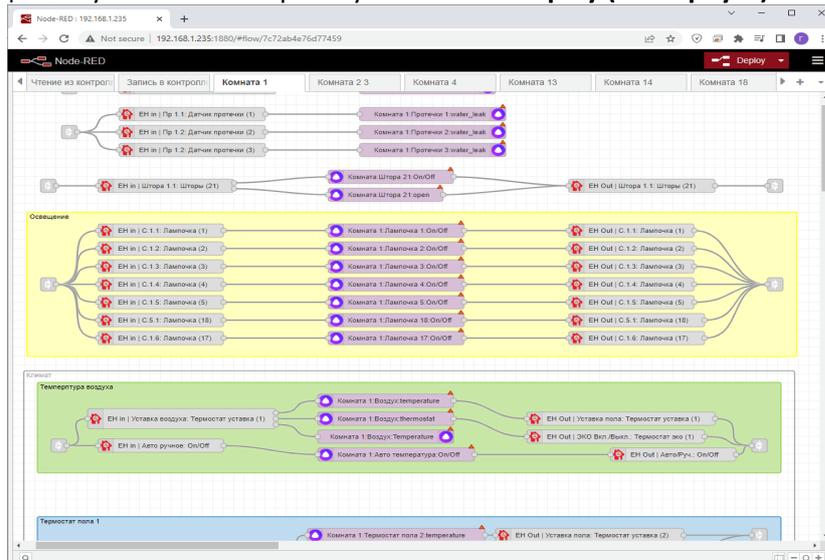


### 7.3. Проверка наличия установленных пакетов в закладке **Nodes (Узлы)**

- 7.3.1. @bitpoolos/edge-modbus
- 7.3.2. node-red
- 7.3.3. node-red-contrib-alice
- 7.3.4. node-red-contrib-buffer-parser
- 7.3.5. node-red-contrib-ehome
- 7.3.6. node-red-contrib-modbus

## 8. Импорт страниц потоков (Flows) в Node-RED

Нажимаем **MENU -> Import -> Examples (Примеры) -> node-red-contrib-ehome -> templates -> main-tmp** (или другой пример или выбираем файл с потоками \*.json с объектами и настройками для данного объекта), загружаются потоки, подтверждаем развёртывание потоков. В отдельные страницы потоков вынесены **Чтение из контроллера** и **Запись в контроллер**, остальные страницы потоков содержат объекты контроллера связанные с объектами Яндекс-Дом и сгруппированы в разные страницы по типам систем или по помещениям. Для запуска потоков в работу нажимаем **Deploy (Развернуть)**.



**На этом этапе закончена предварительная подготовка коммуникационного сервера связи EasyHome с Node-RED и Yandex-Дом. Дальнейшая настройка требует настройки связи с контроллером и организацию объектов связи системы для конкретного проекта.**

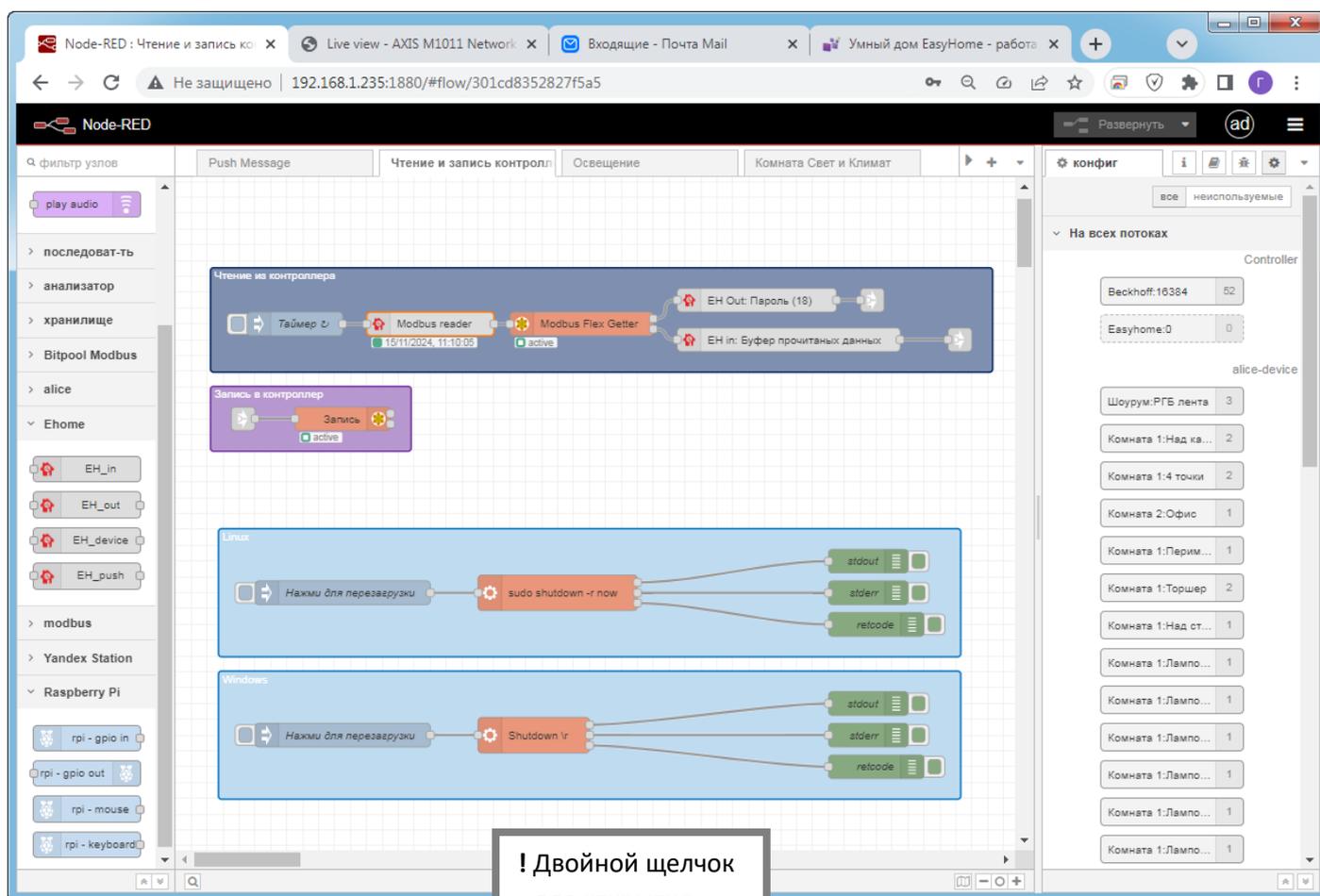
## 2. Потоки чтения и записи в контроллер

В браузере на ПК или на Raspberry открываем интерфейс Node-RED указывая в строке адреса <http://192.168.1.235:1880/> (Пароль по умолчанию **admin 88888888**)

В данной реализации применён протокол связи с контроллером ModbusTCP и поток **Чтение из контроллера** производит периодическое чтение 4х областей памяти контроллера со всеми основными объектами системы EasyHome: 1) Освещение, 2) Климат (30 зон воздух+пол), 3) различные подсистемы, 4) системный запрос. В свойствах устройства **EASYHOME PLC** (узел Device) указаны параметры запросов ModbusTCP. В свойствах узла **EasyHome Connection** (из-modbus-client) можно изменить IP адрес опрашиваемого контроллера и интенсивность опроса.

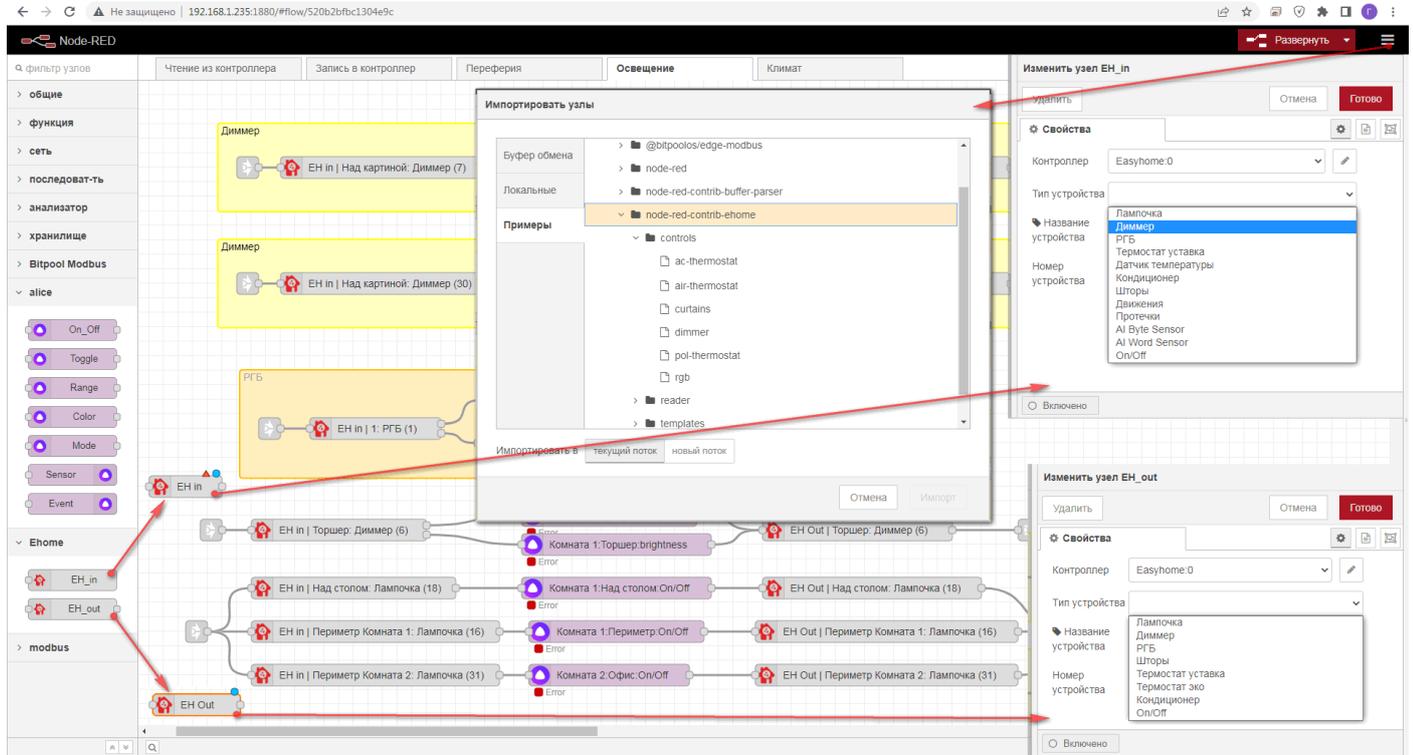
При использовании контроллера **BECKHOFF PLC** надо:

- 2.1. Заменить настройку узла Device из пакета **MENU -> Import -> Examples (Примеры) -> node-red-contrib-ehome -> reader -> reader-bechhof-plc (или reader-ehome-plc)**.
- 2.2. Сменить тип контроллера через в свойства одного любого узла **EH\_in -> Controller -> Controller node -> Beckhoff**, он будет заменён для всех EH\_in и EH\_out. (см. Подробнее в п. 3.2)



### 3. Объекты связи Node-RED с контроллером

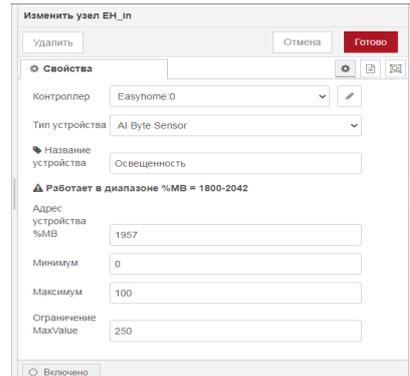
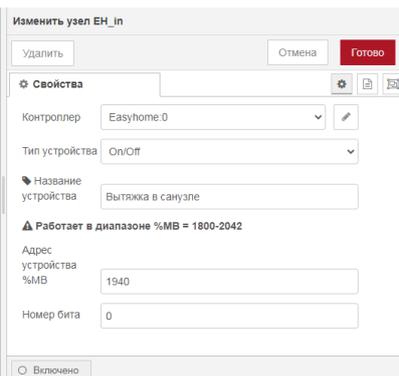
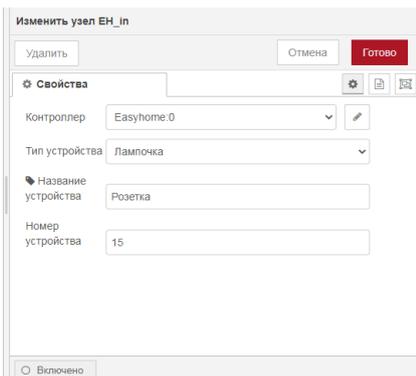
В установленном в Node-RED пакете **node-red-contrib-ehome** присутствуют объекты: **Link\_IN** и **Link\_OUT** (из раздела палитры **Общие**), **EH\_in** и **EH\_out** – чтение и запись данных в контроллер (из раздела палитры **Ehome**), **alice** - для связи с сервером Yandex-Дом (из раздела палитры **alice**):



#### 3.1. EH\_in - ввод данных с контроллера EasyHome – универсальный объект имеющий 11 видов.

Для каждого вида имеются настройки номера или адреса в структуре памяти контроллера EasyHome:

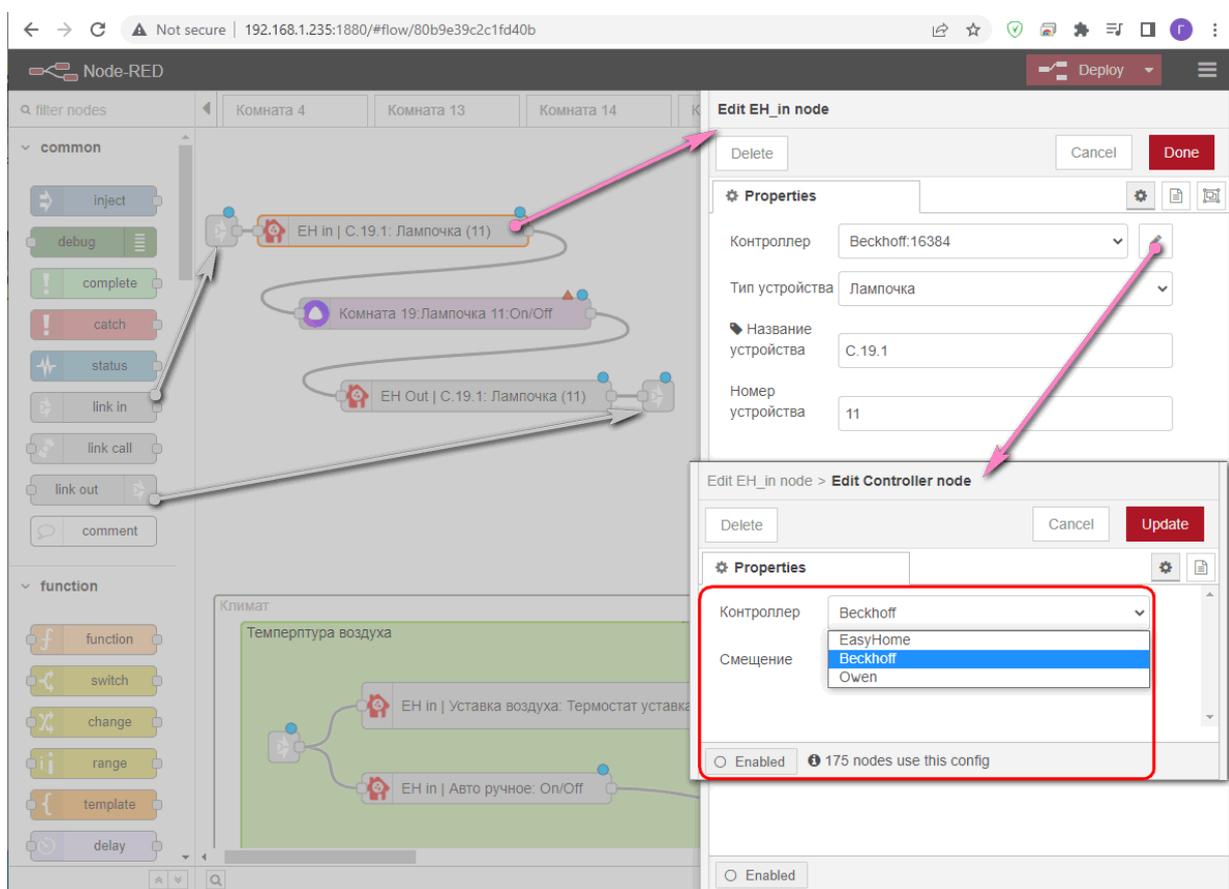
- 3.1.1. Лампочка (задаётся номер лампы 1-255)
- 3.1.2. Диммер (задаётся номер лампы 1-140)
- 3.1.3. РГБ (задаётся номер первой лампы 1-140)
- 3.1.4. Термостат уставка (задаётся номер термостата 1-60, 30 воздух + 30 пол)
- 3.1.5. Датчик температуры (задаётся номер термостата 1-60, 30 воздух + 30 пол)
- 3.1.6. Кондиционер (задаётся номер зоны климата 1-30)
- 3.1.7. Шторы (задаётся номер лампы 1-140)
- 3.1.8. Движение (задаётся номер датчика 1-64)
- 3.1.9. Протечки (задаётся номер датчика 1-32)
- 3.1.10. AI Byte Sensor – произвольный аналоговый/числовой индикатор (адрес %MB1800..2042)
- 3.1.11. AI Word Sensor – произвольный аналоговый/числовой индикатор (адрес %MB1800..2042)
- 3.1.12. On/Off – чтение бита данных (адрес %MB1800..2042.0..7)
- 3.1.13.



3.2. **EH\_out** – вывод данных на контроллер EasyHome – универсальный объект имеющий 8 видов.  
 Аналогично **EH\_in**, для каждого вида имеются настройки номера или адреса в структуре памяти контроллера EasyHome:

- 3.2.1. Лампочка (задаётся номер лампы 1-255)
- 3.2.2. Диммер (задаётся номер лампы 1-140)
- 3.2.3. РГБ (задаётся номер первой лампы 1-140)
- 3.2.4. Шторы (задаётся номер лампы 1-140)
- 3.2.5. Термостат уставка (задаётся номер термостата 1-60, 30 воздух + 30 пол)
- 3.2.6. Термостат эко (задаётся номер термостата 1-60, 30 воздух + 30 пол)
- 3.2.7. Кондиционер (задаётся номер зоны климата 1-30)
- 3.2.8. On/Off – запись бита данных (Задаётся адрес и %MB1800..2042 и номер бита 0..7)

Во всех узлах **EH\_in** и **EH\_out** есть ссылка на общий узел контроллера - **Controller node**, там выбирается нужный тип контроллера EasyHomePLC или Beckhoff PLC или OwenPLC для использования подходящего смещения памяти в запросах ModbusTCP.

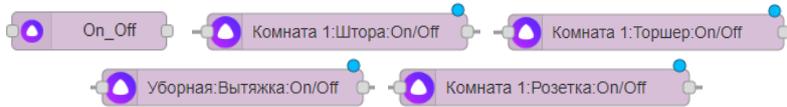


### 3.3. alice - объекты для связи с сервером Яндекс -Дом и Алисы (alice).

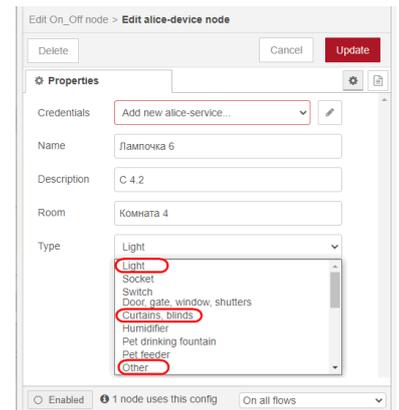
Объекты **Яндекс -Дом** имеют много режимов и подрежимов, подробно описаны на сайте **Yandex** в разделе для разработчиков. Но проще подбирать нужный тип прямо из конфигуратора **Node-RED** и смотреть результат вэб-интерфейсе **Яндекс-Дом** (<https://yandex.ru/iot/>) или в приложении для мобильных устройств. При добавлении или изменении объекта **alice**, после нажатия кнопки **Deploy (Развернуть)**, обновления на интерфейсах **Яндекс-Дом** происходят автоматически в течение нескольких десятков секунд.

При добавлении данных объектов надо учитывать базовое название объекта **Name** и базовое расположение объекта **Room**, так как в вэб-интерфейсе **Яндекс-Дом** и в приложении для мобильных устройств они будут названы соответствующим образом и расположены по соответствующим комнатам. В дальнейшем, названия объектов и комнат и расположение пользователь может менять самостоятельно в вэб-интерфейсе и приложении. Это может потребоваться для лучшего голосового взаимодействия с голосовым ассистентом Алиса.

#### 3.3.1. On\_Off node



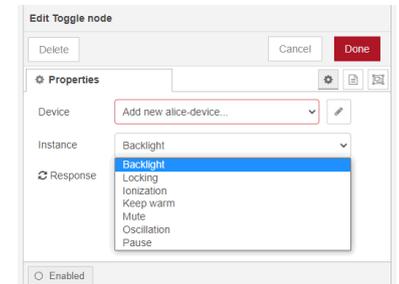
- включение и выключение объекта, тип объекта выбирается.
- используется для Лампочек, Штор и других объектов имеющих функцию Включения и Выключения.



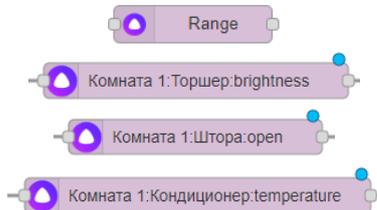
#### 3.3.2. Toggle node



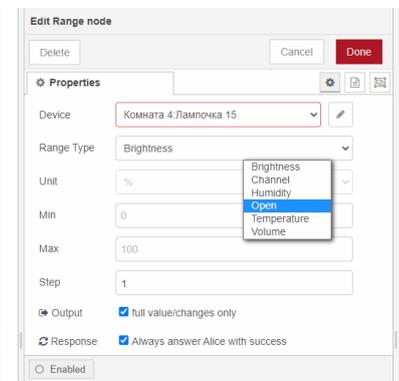
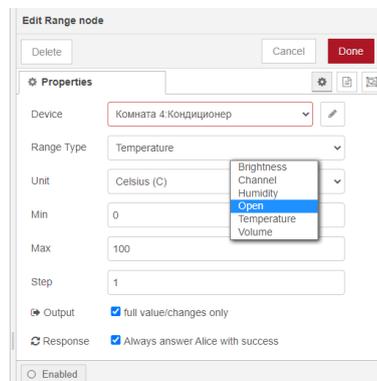
- переключение объекта
- В типовых примерах интеграции EasyHome и Yandex не используется, но может быть полезен в различных реализациях



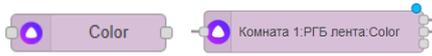
#### 3.3.3. Range node



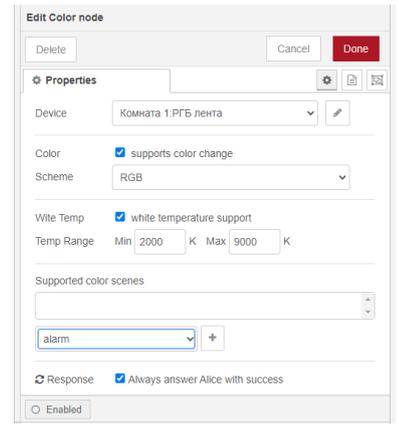
- индикатор и настройка числового значения
- Тип, диапазон и единицы измерения выбираются



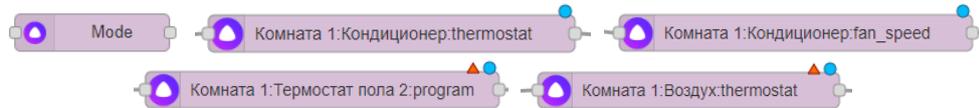
### 3.3.4. Color node



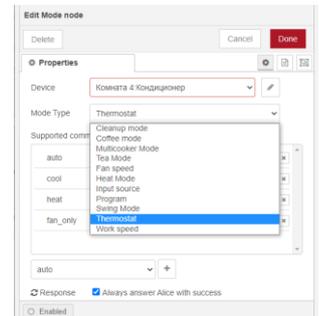
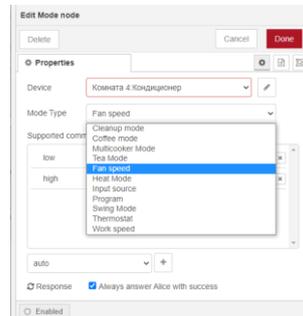
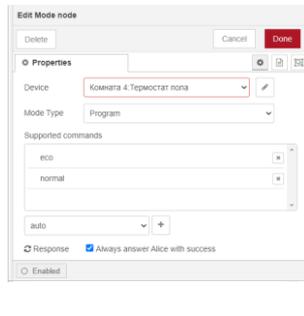
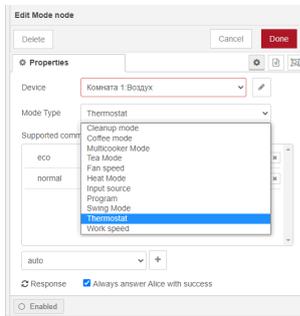
– управление цветом



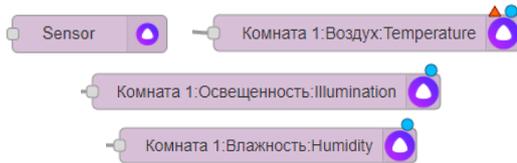
### 3.3.5. Mode node



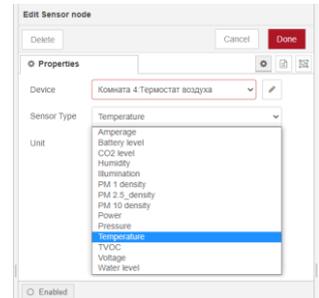
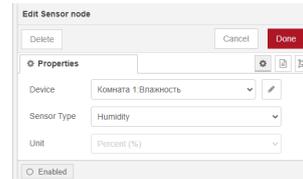
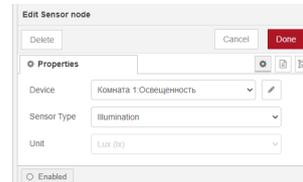
– индикатор и настройка режима по названию, различные списки к разным объектам



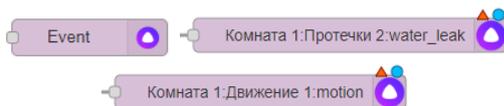
### 3.3.6. Sensor node



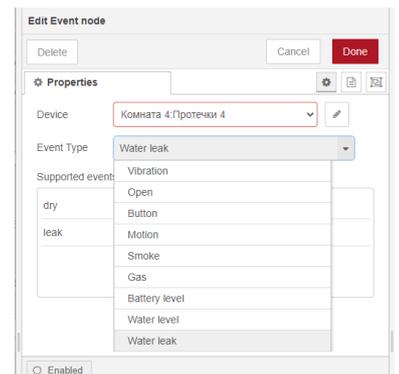
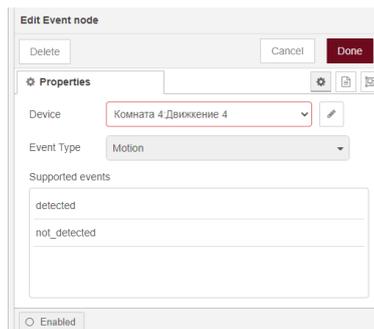
– индикатор состояния датчика с числовым сигналом



### 3.3.7. Event node

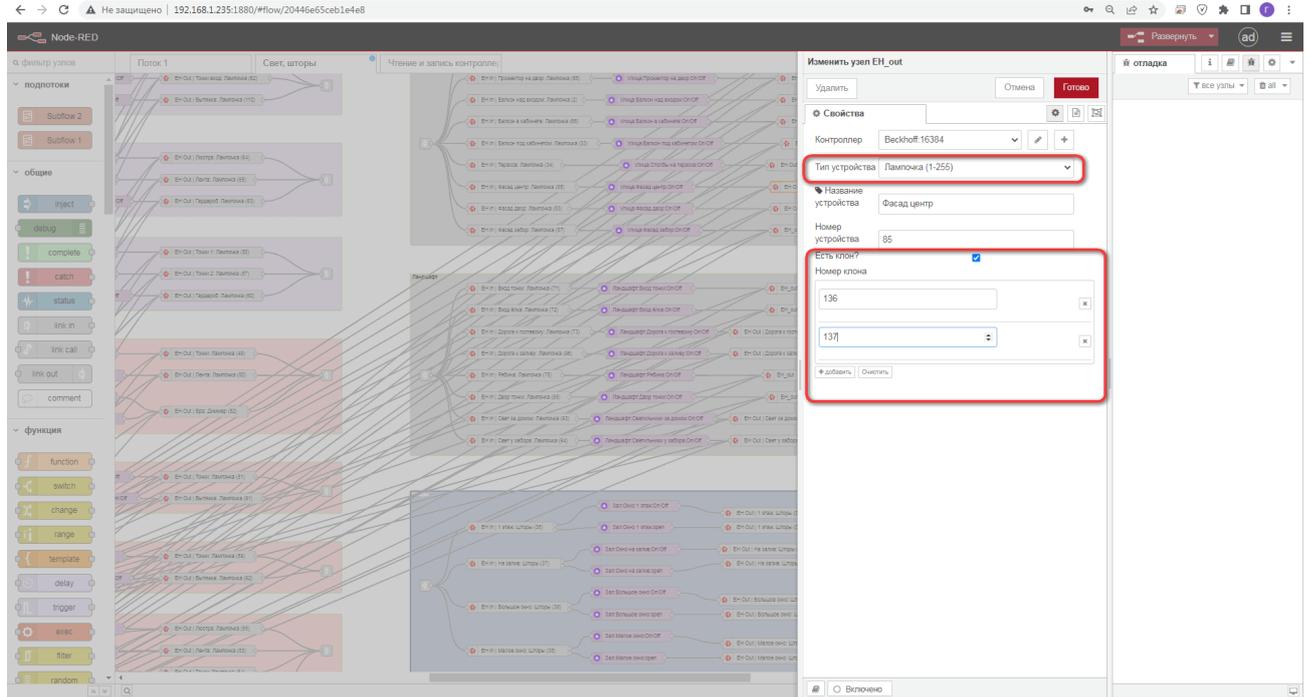


– индикатор события



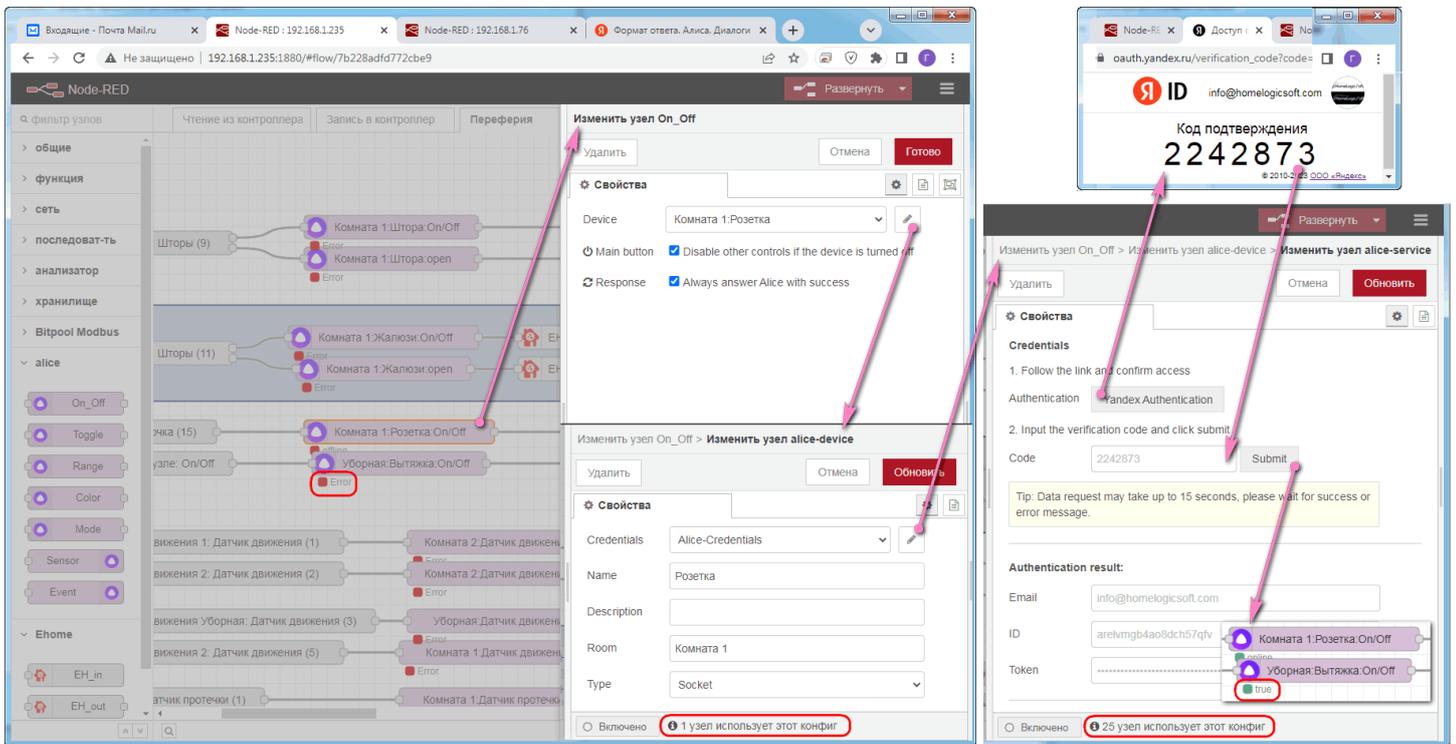
### 3.3.8. Тип контроля ЛАМПА

Для контролов типа ЛАМПА в контроллере существует механизм создания ламп-клонов, которые включатся-выключаются вместе. Такие лампы необходимо указывать и при настройке в NodeRED:



### 3.4. Авторизация объектов alice (Яндекс-Дом)

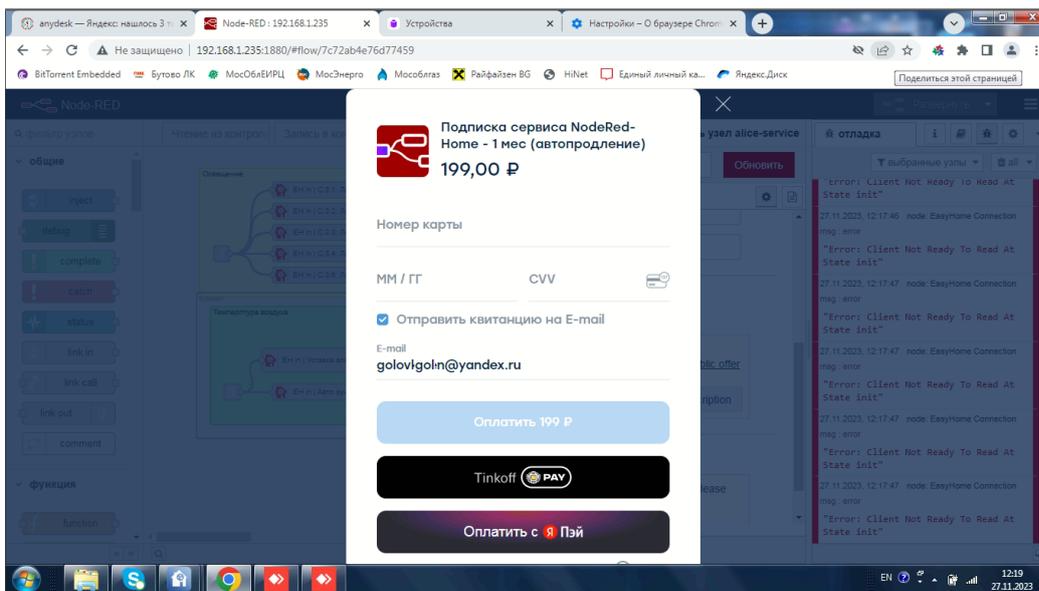
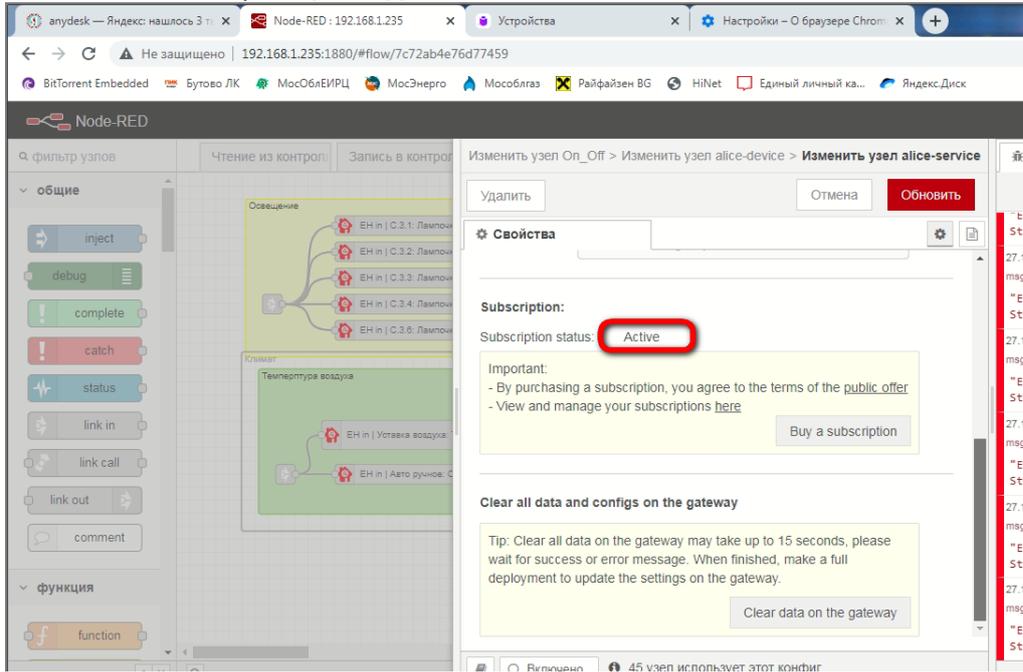
Объекты Яндекс-Дом требуют авторизацию в учётной записи пользователя в Яндексе. При добавлении новых объектов, как правило, авторизация проходит автоматически по уже введённым учётным данным для первого введённого объекта. Рядом с объектом есть индикатор – красный квадратик (нет авторизации) или зелёный и значение объекта.



При добавлении объектов **alice** в Node-RED они добавляются в подключенный образ Яндекс-ДОМ автоматически, через несколько секунд, лучше перезапустить программу визуализации Яндекс-ДОМ.

### 3.5. Оплата сервиса NodeRED-Yandex.DOM(alice-service)

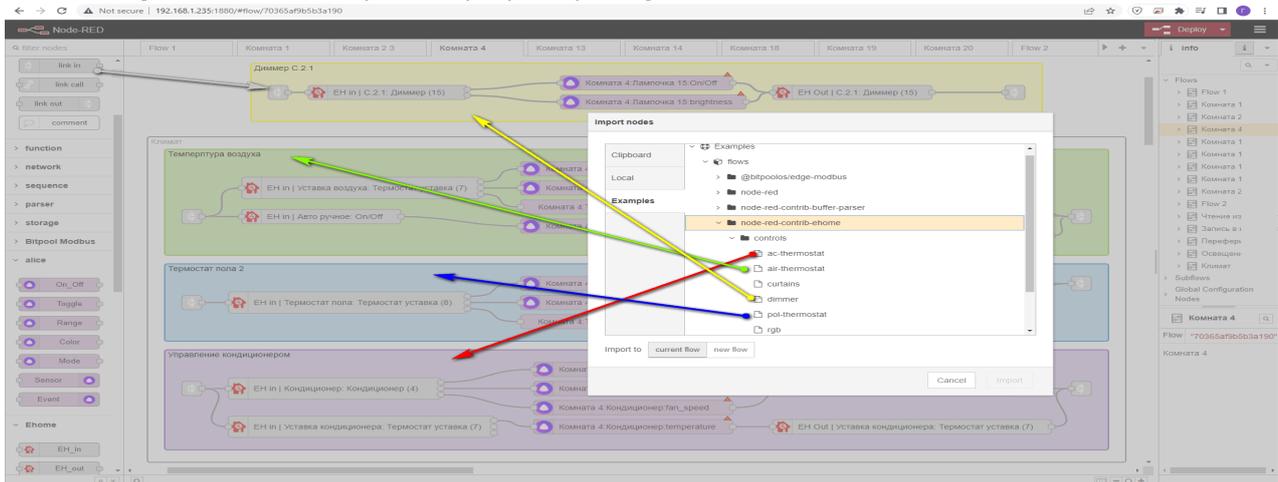
С 2023 сервис взимает плату около 2 евро в месяц, увидеть статус и период оплаты и оплатить сервис можно в окне авторизации Яндекс-объектов:



### 3.6. Комбинированные объекты для сложных устройств доступны в **MENU** -> **Import** -> **Examples** (Примеры) -> **node-red-contrib-ehome** -> **controls**

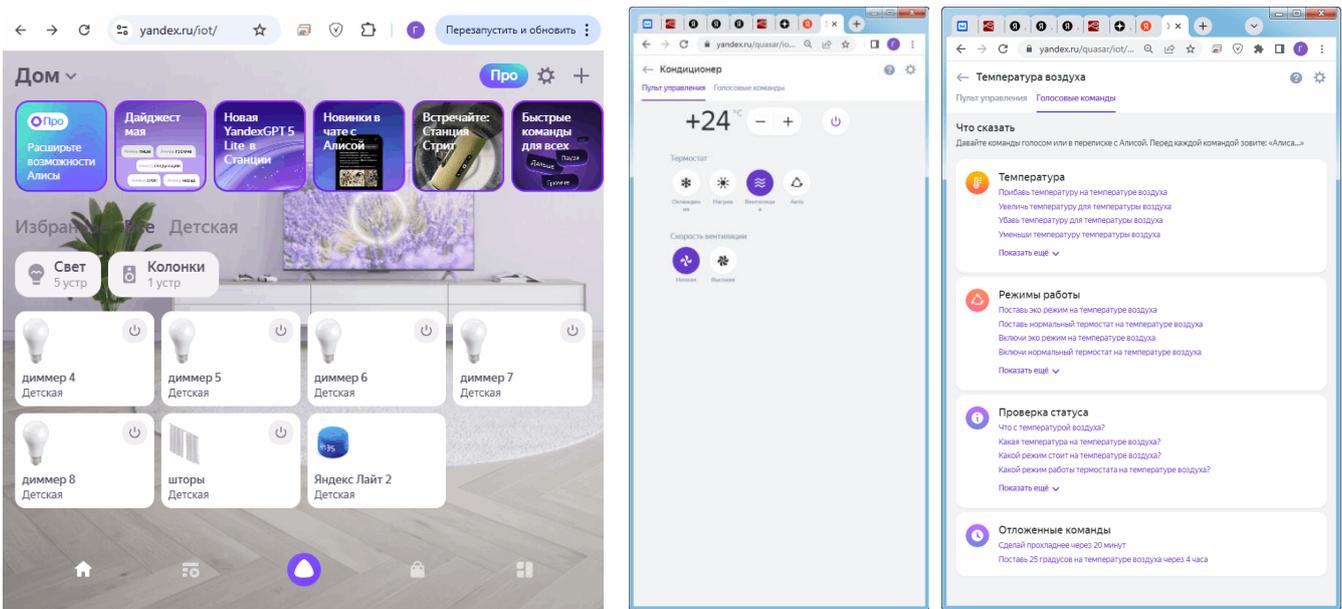
Некоторые объекты требуют обработки сразу нескольких переменных состояния, для них сделаны заготовки:

- 3.6.1. **ac-thermostat** - термостат воздуха, температура воздуха, переключатель ВКЛ-ЭКО и АВТО-РУЧН
- 3.6.2. **air-thermostat** - термостат кондиционера с переключателем режима и скорости
- 3.6.3. **curtains** - регулятор Range и включатель-выключатель OnOff
- 3.6.4. **dimmer** - регулятор Range и включатель-выключатель OnOff
- 3.6.5. **pol-thermostat** - термостат воздуха, температура воздуха, переключатель ВКЛ-ЭКО
- 3.6.6. **rgb** - тройной регулятор Range и включатель-выключатель OnOff



## 4. Вэб-визуализация Яндекс-ДОМ

Для визуализации Яндекс-Дом разработан вэб-интерфейс **Яндекс-Дом** (<https://yandex.ru/iot/>), при открытии страницы через любой браузер требует авторизацию в системе Яндекс. Эта страница удобна для просмотра состояние среды Яндекс-ДОМ с компьютера во время отладки и при проверке связей между EasyHome и Яндекс-Дом установленных в Node-RED :



Вэб-интерфейс и приложение позволяет добавлять устройства, создавать несколько домов к **Яндекс-Дом**, давать гостевой доступ доверенным пользователям, создавать сценарии выполняемые ресурсами **Яндекс-Дом**, группировать объекты и многое другое.

## 5. Визуализация в приложении Яндекс-ДОМ

5.1. Установите приложение

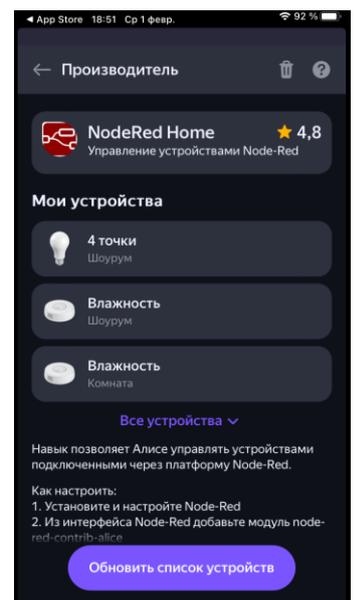
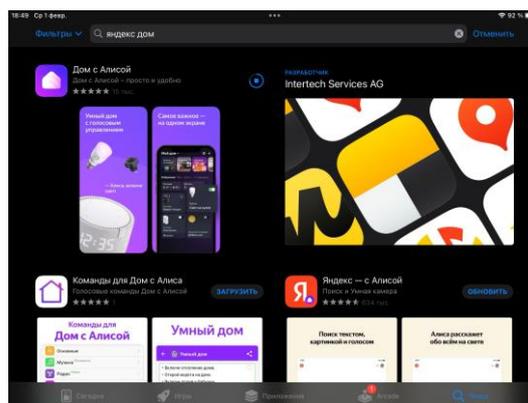
**Дом с Алисой** с AppStore или GooglePlay

5.2. Добавьте устройство **NodeRed\_Home**.

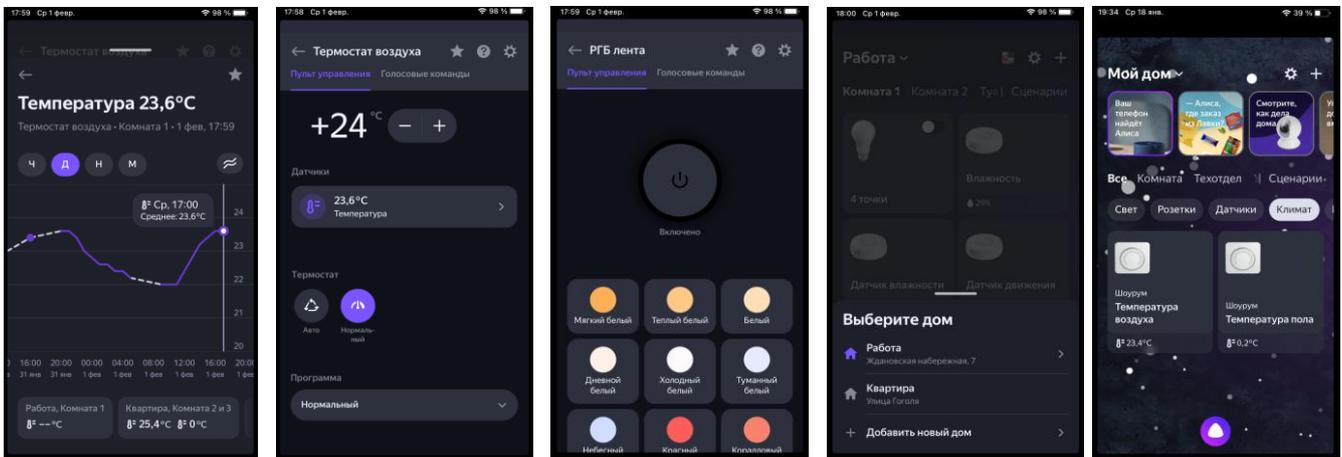
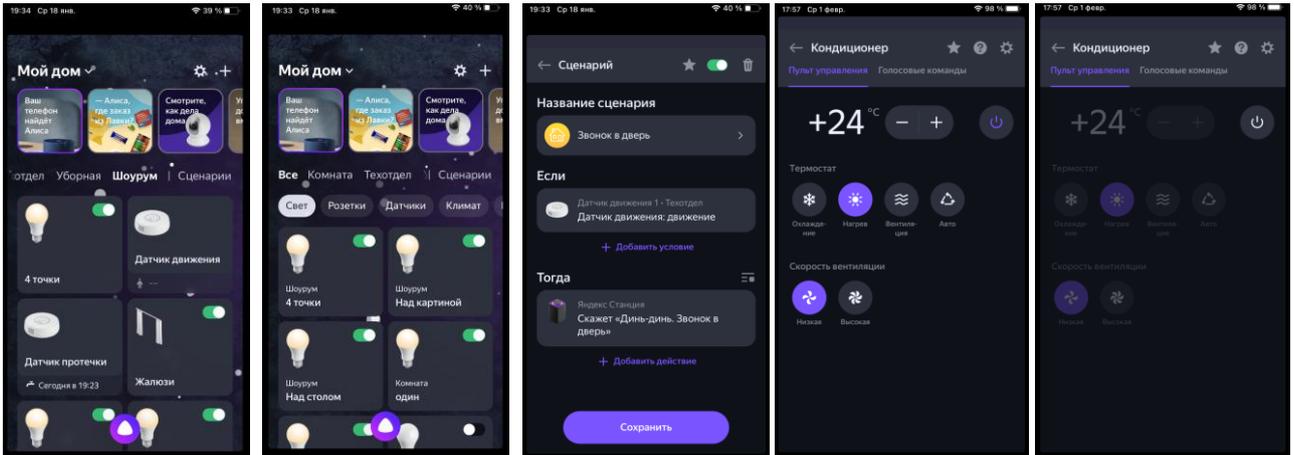
5.3. Введите запрошенную авторизацию в системе Яндекс для устройства Node-RED.

5.4. Из шлюза NodeRed\_Home будет импортирован список всех устройств **alice** с учётом из названия (Name) и расположения (Room).

5.5. Готово к использованию! Голосовое управление будет работать со всех устройств с пройденной авторизацией и включенным голосовым помощником Алиса.

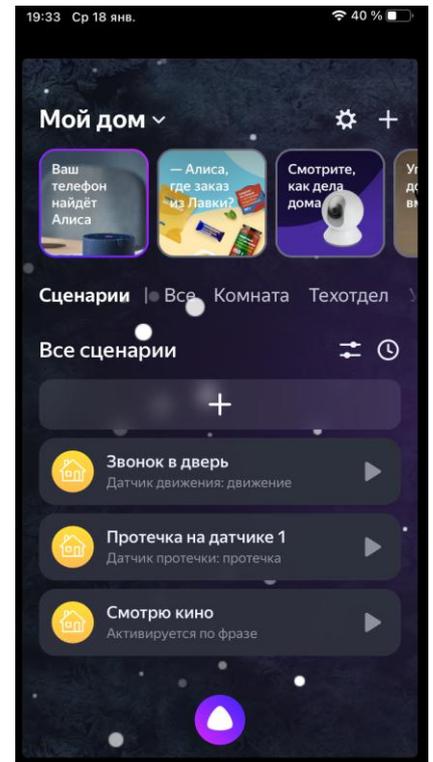


5.6. Из приложения и веб-интерфейса можно **изменить названия устройств, комнат и расположение** по комнатам и домам. Скорее всего, это понадобится сделать для комфортного взаимодействия пользователя с голосовым помощником.

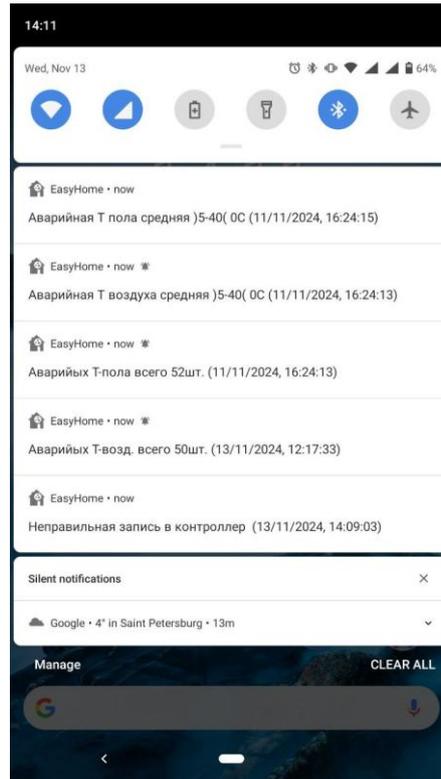


5.7. В веб-версии и в приложении есть закладка с подсказками **Голосовые команды**, но там приведены только базовые примеры, лингвистические возможности Алисы намного шире.

5.8. В веб-версии и в приложении есть возможность создавать **Сценарии автоматизации**, выполняемые средой Яндекс-Дом.



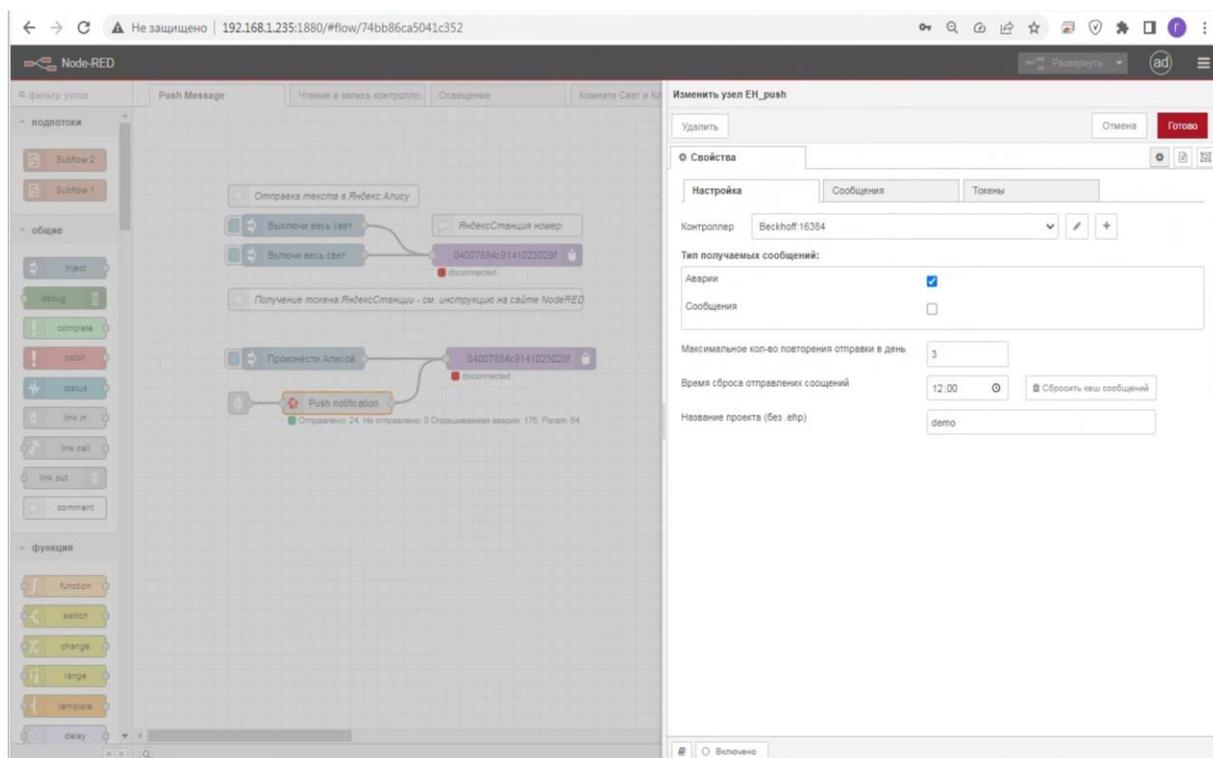
## 6. Настройка PUSH-уведомлений



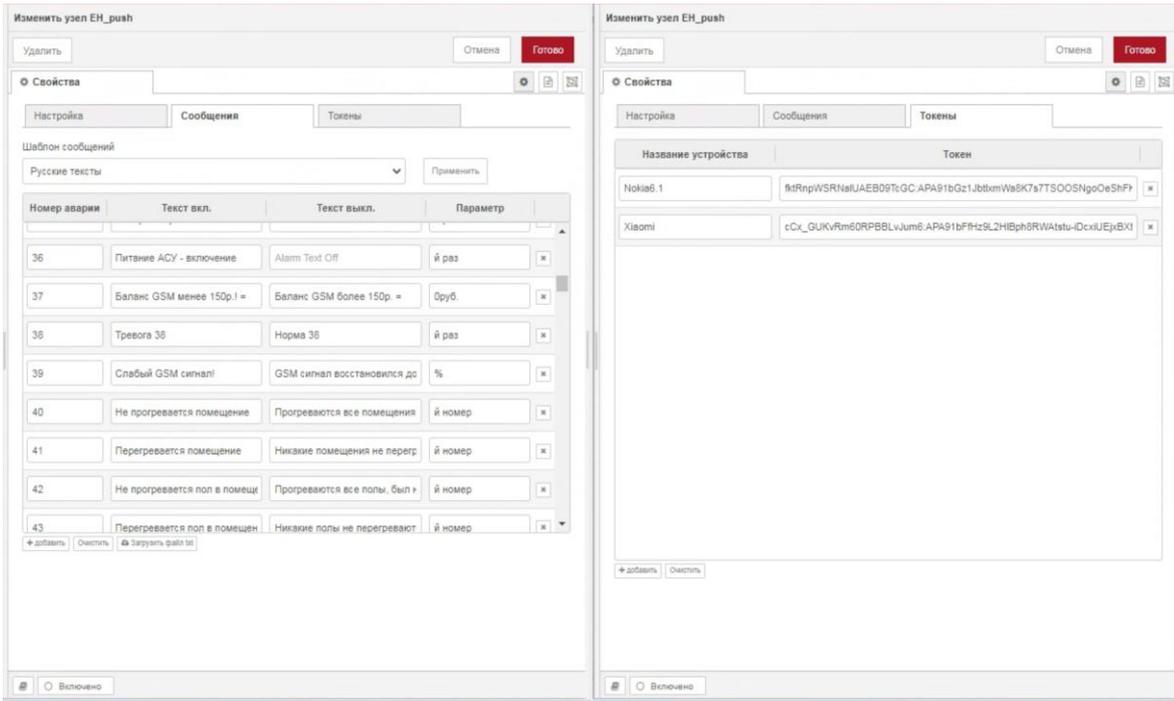
Для работы блока PUSH уведомлений требуется NodeRED4 с nodejs20 или свежее (см.п.1.4 по обновлению и для самостоятельной настройки).

### 6.1. Настройка PUSH уведомлений

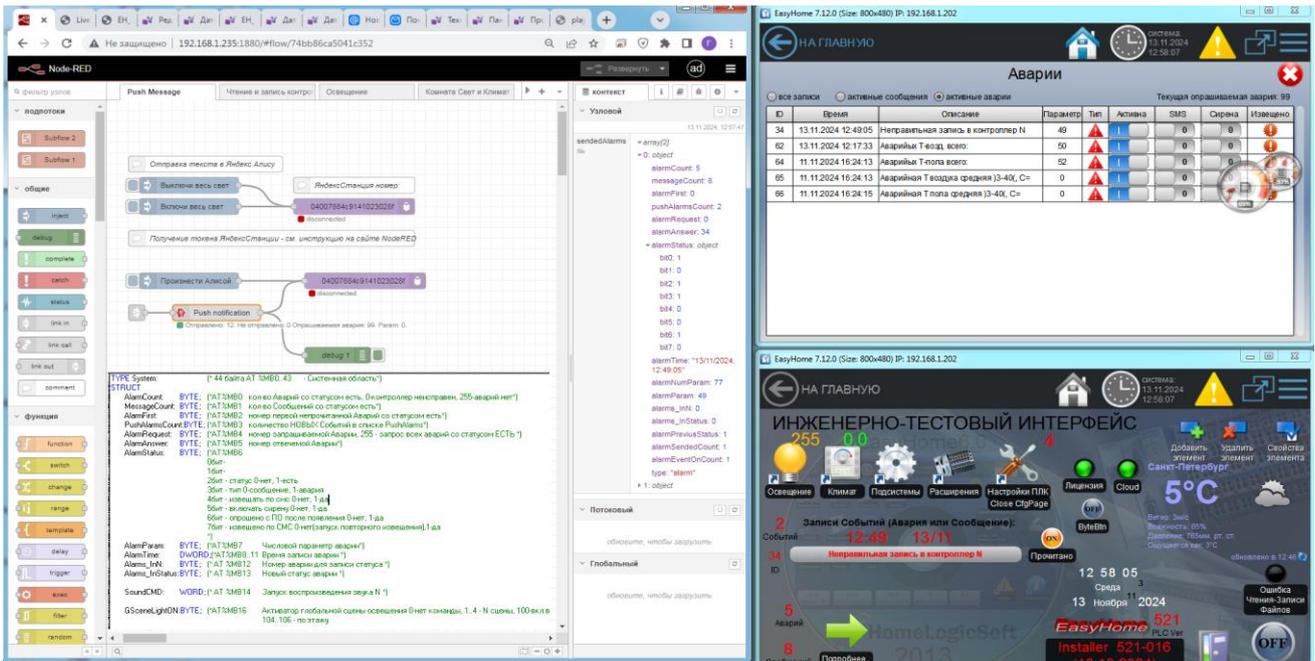
Настройка текстов и правил рассылки аналогична рассылкам SMS-уведомлений и для инициализации массива используются файлы SMS и E-MAIL сообщений SMS\_ru.txt или SMS\_en.txt. После начальной загрузки сообщений из файлов их можно редактировать и настраивать вручную через WEB-интерфейс NodeRED:



Тут же указывается название для поиска проекта EasyHome для запуска при нажатии на уведомление.



Тексты можно загрузить из стандартных файлов и создавать, редактировать, удалять вручную.



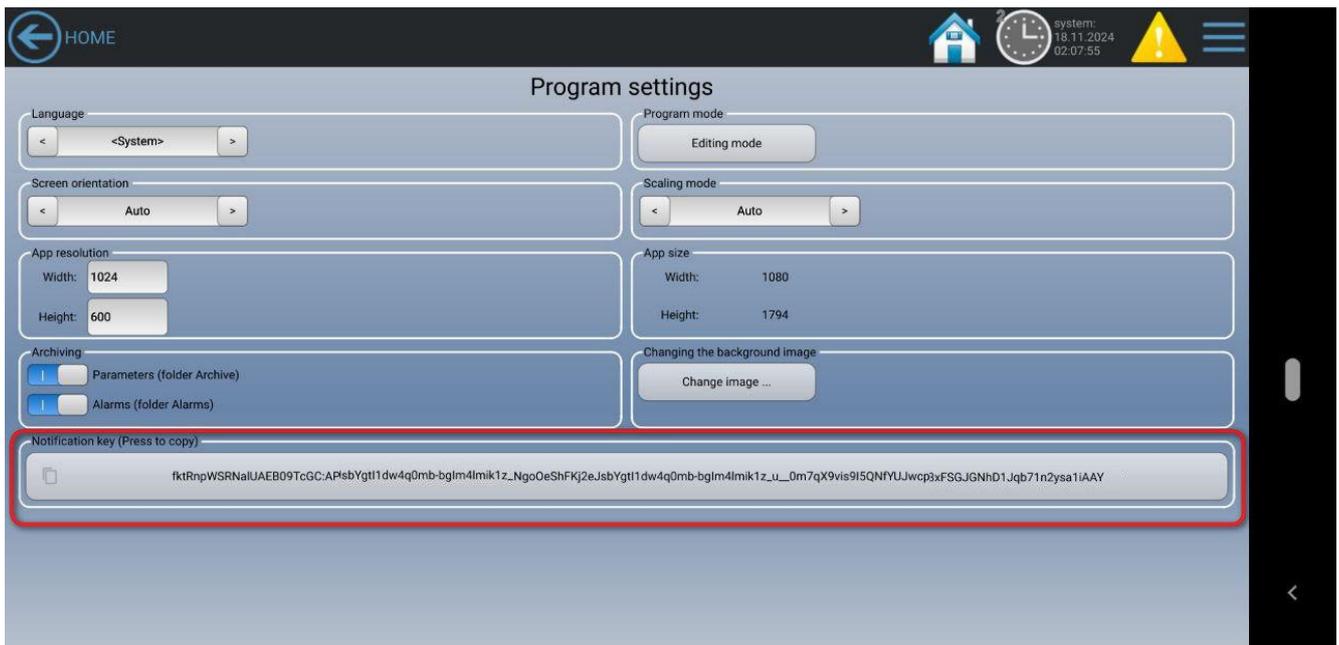
Статусы прочитанных и отосланных сообщений отображаются под блоком PUSH. В разделе КОНТЕКСТ->Узловой можно увидеть массив хранящихся сообщений и их параметров (сохраняются только те, которые хоть раз были активны).

Из блока PUSH выходит текст, который можно направлять в любые другие блоки для отсылки, архивирования, озвучивания и т.д.

## 6.2. Привязка пользовательских устройств

Для рассылки PUSH уведомлений используются штатные сервисы производителей ОС для мобильных устройств Apple (iOS) и Google (Android). Для привязки пользовательского приложения на конкретном мобильном устройстве используются генерируемые при установке приложения токены UUID. В ПО

пользовательского интерфейса **EasyHome** начиная с версии **7.13** этот токен можно увидеть и скопировать со страницы НАСТРОЙКИ:



### 6.3. Настройка времени в Raspberry Pi

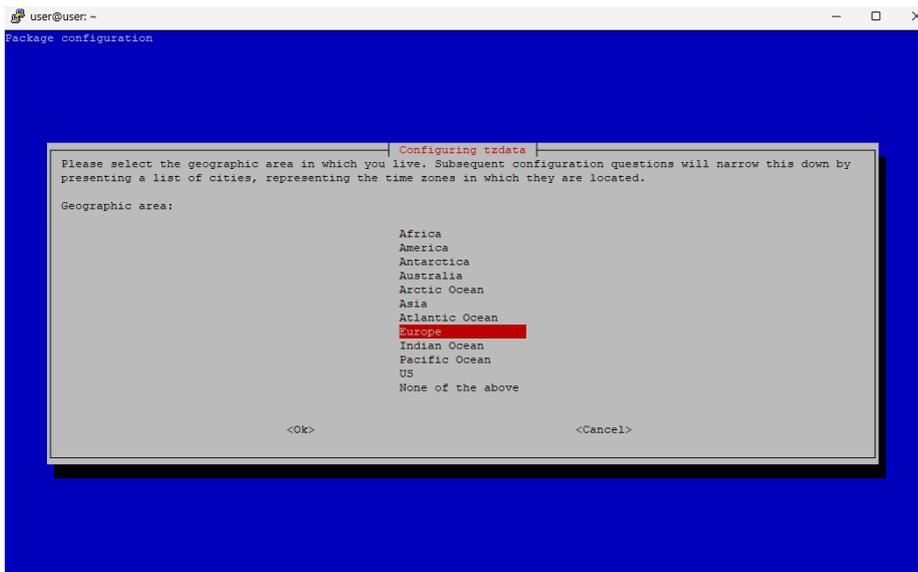
При поставке готового EasyHomeCS часовой пояс настроен на Мск или по заказу. При самостоятельной настройке настроить часы и часовой пояс, так как время сообщения отправляется в уведомлении и по времени осуществляется сброс дневного лимита на повторы одинаковых уведомлений. Настроить время на Raspberry Pi можно через графическую оболочку пользователя и монитор с клавиатурой или через консоль SSH. Команда запуска меню настроек **sudo dpkg-reconfigure tzdata**, команда проверки времени **date**:

```

user@user: ~
user@user:~$ sudo dpkg-reconfigure tzdata

Current default time zone: 'Europe/Moscow'
Local time is now:      Mon Nov 11 17:21:50 MSK 2024.
Universal Time is now:  Mon Nov 11 14:21:50 UTC 2024.

user@user:~$ date
Mon 11 Nov 17:22:13 MSK 2024
user@user:~$
  
```



### 6.4. Настройка сохранения статуса PUSH уведомлений

При поставке готового EasyHomeCS эта функция включена. При самостоятельно настройке для сохранения значений счётчиков отправленных PUSH уведомлений (архив "sendedAlarms") рекомендуется включить сохранение файла с архивом на microSD карту, для этого надо открыть скрытый каталог **/home/user/.node-red** и в файле **settings.js** в 428 строке, после **editorTheme** указать:

```
contextStorage: {
  default: "memoryOnly",
  memoryOnly: { module: 'memory' },
  file: { module: 'localfilesystem' }
},
```

В противном случае счётчики отправленных сообщений будут сбрасываться при каждой перезагрузке Raspberry и каждой активации обновлений проекта в NodeRED кнопкой **Deploy (Развернуть)**.

