

Gate System Requirements - Home Assistant Version

Файлы для GateCheck.py (большие ворота):

Python файлы:

- GateCheck.py - основная программа мониторинга больших ворот
- ControlSwitch.py - управление Shelly реле для открытия/закрытия ворот
- TelegramButtonsGen.py - модуль для отправки Telegram уведомлений с кнопками

Конфигурация:

- gate_check.ini - конфигурационный файл

Обязательные секции в gate_check.ini:

ini

[Telegram ID]

TOKEN = "your_telegram_bot_token"

chat_id = "your_chat_id"

[HA]

HA_IP = "192.168.2.43"

HA_TOKEN = "your_home_assistant_token"

big_gate_opening_entity = "binary_sensor.lumi_lumi_sensor_magnet_aq2_opening_2"

[Device ID]

ip_gate = 192.168.2.141 # IP адрес Shelly реле

[Time-outs]

time_polling = 180

time_to_close = 120

close_tries = 3

delay_1 = 5

delay_2 = 15

delay_3 = 30

[Battery limits]

battery_limit_1 = 15

battery_limit_2 = 5

Файлы для GateCheckSmall.py (малые ворота):

Python файлы:

- `GateCheckSmall.py` - основная программа мониторинга малых ворот
- `TelegramButtonsGen.py` - модуль для отправки Telegram уведомлений с кнопками

Конфигурация:

- `gate_check.ini` - тот же конфигурационный файл (общий)

Используемые секции в `gate_check.ini`:

ini

[Telegram ID]

TOKEN = "your_telegram_bot_token"

chat_id = "your_chat_id"

[HA]

HA_IP = "192.168.2.43"

HA_TOKEN = "your_home_assistant_token"

small_gate_opening_entity = "binary_sensor.lumi_lumi_sensor_magnet_aq2_opening"

[Time-outs]

time_polling_small = 20

time_to_close_small = 10

delay_1_small = 1

delay_2_small = 5

delay_3_small = 15

delay_4_small = 60

[Battery limits]

battery_limit_1 = 15

battery_limit_2 = 5

Файлы для `Online_check_web.py` (веб-интерфейс):

Python файлы:

- `Online_check_web.py` - основная веб-программа
- `ControlSwitch.py` - управление Shelly реле
- `TelegramButtonsGen.py` - уведомления для alert-сообщений

Веб-файлы:

- `templates/status.html` - HTML шаблон веб-интерфейса
- `static/` - папка для CSS/JS файлов (опционально)

Конфигурация:

- `online_check.ini` - конфигурационный файл

Обязательные секции в `online_check.ini`:

ini

[Computers]

Server = 192.168.2.136 RPI 30 10

FR-Ext = 192.168.2.44 RPI 30 10

HomeAssistant = 192.168.2.43 RPI 30 10

... другие устройства

[Sensors]

BigGate = DUMMY SENSOR 180 120 5 10 10

SmallGate = DUMMY SENSOR 180 120

ip_gate = 192.168.2.141

[HA]

HA_IP = "192.168.2.43"

HA_TOKEN = "your_home_assistant_token"

big_gate_opening_entity = "binary_sensor.lumi_lumi_sensor_magnet_aq2_opening_2"

small_gate_opening_entity = "binary_sensor.lumi_lumi_sensor_magnet_aq2_opening"

[Telegram ID]

TOKEN = "your_telegram_bot_token"

chat_id = "your_chat_id"

[Time-outs]

delay_1 = 0

delay_2 = 1

delay_3 = 5

delay_4 = 60

delay_telegram = 1

Python зависимости:

Установка через `pip`:

Общие для обеих программ:

bash

`pip install requests`

`pip install python-telegram-bot`

Дополнительно для веб-программы:

bash

pip install fastapi

pip install uvicorn

pip install jinja2

pip install sse-starlette

Или установка одной командой:

bash

pip install requests python-telegram-bot fastapi uvicorn jinja2 sse-starlette

Структура файлов и папок:

/GateCheck/

— GateCheck.py	# Программа мониторинга больших ворот
— GateCheckSmall.py	# Программа мониторинга малых ворот
— Online_check_web.py	# Веб-интерфейс мониторинга
— ControlSwitch.py	# Управление Shelly реле (только для больших ворот)
— TelegramButtonsGen.py	# Telegram уведомления
— gate_check.ini	# Общая конфигурация для GateCheck.py и GateCheckSmall.py
— online_check.ini	# Конфигурация для веб-интерфейса
— templates/	
— status.html	# HTML шаблон веб-интерфейса
— static/	# Статические файлы (опционально)

Системные требования:

Home Assistant:

- Работающий экземпляр Home Assistant
- Настроенные датчики ворот (binary_sensor и sensor для батареи)
- Long-lived access token

Telegram Bot:

- Созданный Telegram бот через @BotFather
- Bot token
- Chat ID для отправки уведомлений

Сеть:

- Shelly реле для управления воротами

- Настроенный брандмауэр Windows (порт 8000 для веб-интерфейса)
-



Запуск программ:

GateCheck.py (большие ворота):

```
bash
```

```
python GateCheck.py
```

GateCheckSmall.py (малые ворота):

```
bash
```

```
python GateCheckSmall.py
```

Online_check_web.py (веб-интерфейс):

```
bash
```

```
python Online_check_web.py
```

Веб-интерфейс будет доступен по адресу:

Примечание: Все три программы могут работать одновременно, так как каждая выполняет свою функцию.



Проверка готовности:

Для GateCheck.py (большие ворота):

- ☐ 3 Python файла в папке (GateCheck.py, ControlSwitch.py, TelegramButtonsGen.py)
- ☐ Файл gate_check.ini с правильными секциями
- ☐ Home Assistant доступен и настроен
- ☐ Telegram бот создан и настроен
- ☐ Shelly реле доступно по сети

Для GateCheckSmall.py (малые ворота):

- ☐ 2 Python файла в папке (GateCheckSmall.py, TelegramButtonsGen.py)
- ☐ Файл gate_check.ini с правильными секциями (тот же что и для больших ворот)
- ☐ Home Assistant доступен и настроен
- ☐ Telegram бот создан и настроен

Для Online_check_web.py (веб-интерфейс):

- ☐ 3 Python файла в папке (Online_check_web.py, ControlSwitch.py, TelegramButtonsGen.py)
 - ☐ Файл templates/status.html
 - ☐ Файл online_check.ini с правильными секциями
 - ☐ Брандмауэр настроен для порта 8000
 - ☐ Home Assistant доступен и настроен
 - ☐ Telegram бот настроен (для alert-ов)
-

Настройка брандмауэра Windows:

Для доступа к веб-интерфейсу из сети:

1. **Windows Defender Firewall** → **Advanced settings** → **Inbound Rules**
 2. **New Rule...** → **Port** → **TCP** → **8000**
 3. **Allow the connection** → отметить все профили → **Finish**
-

Мобильный доступ:

Веб-интерфейс адаптирован для мобильных устройств и доступен с любого устройства в сети по адресу

Система полностью переведена с TuYa Cloud на локальный Home Assistant для обеспечения независимости от интернета и повышения скорости отклика.