

Umhost Agent: Руководство по установке и эксплуатации

Содержание

1. Введение
 2. Краткое Описание Umhost Agent
 3. Требования системы
 4. Установка Umhost Agent
 - Установка Supervisor
 - Установка Umhost Agent
 5. Настройка Supervisor
 6. Запуск и управление агентом
 7. Подключение к MQTT-Серверу
 8. Использование API Umhost Agent
 - Эндпоинты
 - Модели данных
 9. Примеры Команд
 - Включение устройств из сна
 - Выключение устройств и создание Расписания
 10. Обработка ошибок и Диагностика
 11. Обновление Umhost Agent
 12. Безопасность
-

Введение

Добро пожаловать в руководство пользователя Umhost Agent — мощного инструмента для управления и мониторинга майнинг-устройств в дата-центрах. Это руководство предназначено для помощи пользователям в установке, настройке и эффективном использовании Umhost Agent для оптимизации работы их оборудования.

Краткое Описание Umhost Agent

Umhost Agent — это программное обеспечение, разработанное для централизованного управления майнинг-устройствами в дата-центрах. Он предоставляет следующие возможности:

- **Мониторинг:** Отслеживание состояния и производительности оборудования в реальном времени.
- **Контроль:** Управление питанием устройств, включая включение и выключение.
- **Автоматизация:** Автоматическая настройка и конфигурация устройств.
- **Управление через API и MQTT:** Без графического интерфейса, управление осуществляется через MQTT-сервер или API-команды.
- **Расписание задач:** Планирование задач управления питанием для оптимизации энергопотребления и увеличения срока службы оборудования.

Umhost Agent предназначен для использования на системах с архитектурой ARM64, таких как Raspberry Pi OS или Ubuntu, обеспечивая надежное и стабильное управление большим количеством устройств.

Требования Системы

Перед установкой Umhost Agent убедитесь, что ваша система соответствует следующим требованиям:

- **Операционная система:** Ubuntu (любая современная версия) или Raspberry Pi OS.
 - **Архитектура:** ARM64.
 - **Права доступа:** Root или пользователь с привилегиями sudo.
 - **Дополнительные зависимости:** Supervisor для управления процессами.
 - **Сетевые требования:** Доступ к MQTT-брокеру или возможности для использования API через сеть.
-

Установка Umhost Agent

Процесс установки состоит из нескольких этапов: установка необходимых зависимостей, скачивание и установка Umhost Agent, а также настройка системы для автоматического запуска агента.

Установка Supervisor

Supervisor — это система управления процессами, которая обеспечивает автоматический запуск и перезапуск Umhost Agent.

Обновите список пакетов:

```
sudo apt update
```

Установите Supervisor:

```
sudo apt install supervisor
```

Установка Umhost Agent

Создайте директорию для Umhost Agent:

```
sudo mkdir -p /etc/umhost
```

Скачайте Umhost Agent:

Предполагается, что у вас уже есть файл umhost-linux-arm64. Если нет, свяжитесь с поставщиком или скачайте его из официального источника.

```
sudo cp путь_до_файла/umhost-linux-arm64 /etc/umhost/
```

1. *Примечание: Замените путь_до_файла на фактический путь к вашему файлу.*

Дайте исполняемые права программе:

```
sudo chmod +x /etc/umhost/umhost-linux-arm64
```

Настройка Supervisor

Для обеспечения автоматического запуска и перезапуска Umhost Agent используется Supervisor. Настройка включает создание конфигурационного файла для Supervisor.

Создайте конфигурационный файл Supervisor для Umhost Agent:

```
sudo nano /etc/supervisor/conf.d/umhost.conf
```

Добавьте следующую конфигурацию в файл:

```
[program:umhost]
command=/etc/umhost/umhost-linux-arm64
autostart=true
autorestart=true
stderr_logfile=/var/log/umhost.err.log
stdout_logfile=/var/log/umhost.out.log
```

1. Описание полей:

- `command`: Полный путь к исполняемому файлу Umhost Agent.
- `autostart`: Автоматический запуск агента при старте системы.
- `autorestart`: Автоматический перезапуск агента в случае сбоя.
- `stderr_logfile` и `stdout_logfile`: Пути к файлам журналов для ошибок и стандартного вывода.

2. Сохраните и закройте файл:

- Нажмите Ctrl + O, затем Enter для сохранения.
 - Нажмите Ctrl + X для выхода из редактора.
-

Запуск и Управление Агентом

После настройки Supervisor можно запустить Umhost Agent и управлять им с помощью Supervisor команд.

Запустите агент для создания начальной конфигурации:

```
sudo /etc/umhost/umhost-linux-arm64
```

Примечание: Это может создать необходимые файлы конфигурации или выполнить начальную инициализацию.

Обновите конфигурацию Supervisor:

```
sudo supervisorctl reread  
sudo supervisorctl update
```

reread: Инструктирует Supervisor перечитать конфигурационные файлы.

update: Применяет изменения и запускает новые программы .

Запустите Umhost Agent в фоновом режиме:

```
sudo supervisorctl start umhost
```

Проверьте статус работы агента:

```
sudo supervisorctl status umhost
```

Вы должны увидеть статус RUNNING для агента.

Остановка агента:

```
sudo supervisorctl stop umhost
```

Перезапуск агента:

```
sudo supervisorctl restart umhost
```

Подключение к MQTT-Серверу

Umhost Agent использует MQTT-протокол для получения и обработки команд управления. Для успешного подключения необходимо настроить доступ к MQTT-брокеру.

Требования к MQTT-Серверу

- **Адрес брокера:** IP-адрес или доменное имя вашего MQTT-брокера.
- **Порт:** 1883.
- **Учетные данные:** Логин и пароль, если ваш брокер требует аутентификацию.
- **Топики:** Определите топики для отправки и получения команд, например, umhost/commands.

Пример Подключения

```
mqtt_subscriber --host mqtt.example.com --port 1883 --topic umhost/commands
```

Примечание: Убедитесь, что mqtt_subscriber установлен и настроен на вашем устройстве.

Настройка Umhost Agent для MQTT

Откройте файл конфигурации Umhost Agent (если имеется):

```
sudo nano /etc/umhost/config.json
```

1.

Добавьте или обновите настройки MQTT:

```
{  
  "mqtt": {  
    "host": "mqtt.example.com",  
    "port": 1883,  
    "username": "your_username",  
    "password": "your_password",  
    "topic": "umhost/commands"  
  },  
  ...  
}
```

2.

Сохраните и закройте файл, затем перезапустите агент:

```
sudo supervisorctl restart umhost
```

3.

Использование API Umhost Agent

Umhost Agent предоставляет RESTful API для управления подсетями и планирования задач питания устройств. Это позволяет интегрировать агент с другими системами и автоматизировать процессы.

Эндпоинты

1. Получить Все Подсети

- **URL:** /api/subnetworks
- **Метод:** GET
- **Описание:** Возвращает список всех настроенных подсетей.

Пример Запроса:

```
curl -X GET https://your-server.com/api/subnetworks
```

Пример Ответа:

```
{
  "subnetworks": [
    {
      "id": 1,
      "ip": "10.7.24.1"
    },
    {
      "id": 2,
      "ip": "10.7.23.0/24"
    }
  ]
}
```

2. Получить Расписание Задач

- **URL:** /api/scheduler
- **Метод:** GET
- **Описание:** Возвращает все запланированные задачи управления питанием.

Пример Запроса:

```
curl -X GET https://your-server.com/api/scheduler
```

Пример Ответа (Успех):

```
{
  "jobs": [
    {
      "id": 1,
      "ip_addr": "192.168.1.10",
      "status": 0,
      "power_state": 1,
      "start_time": 1625068800,
      "end_time": 1625072400,
      "timestamp": 1731601221
    },
    {
      "id": 2,
      "ip_addr": "192.168.1.11",
      "status": 0,
      "power_state": 0,
      "start_time": 1625076000,
      "end_time": 1625079600,
      "timestamp": 1731601221
    }
  ]
}
```

Пример Ответа (Задачи Не Найдены):

```
{
  "jobs": null
}
```

3. Создать Расписание Задач

- **URL:** /api/scheduler
- **Метод:** POST
- **Описание:** Создает новые задачи управления питанием. Перед добавлением новых задач все существующие удаляются.

Пример Запроса:

```
curl -X POST https://your-server.com/api/scheduler \
-H "Content-Type: application/json" \
-d '{
  "jobs": [
    {
      "id_ip_addr": 1,
      "password": "password123",
```

```
"power_state": 1,  
"start_time": 1625068800,  
"end_time": 1625072400  
},  
{  
  "id_ip_addr": 2,  
  "password": "password456",  
  "power_state": 0,  
  "start_time": 1625076000,  
  "end_time": 1625079600  
}  
]  
'
```

Пример Ответа (Успех):

```
{  
  "message": "Расписание успешно обновлено"  
}
```

Пример Ответа (Ошибка):

```
{  
  "error": "Данные отсутствуют или некорректны"  
}
```

или

```
{  
  "error": "Ошибка при обновлении расписания"  
}
```

Модели Данных

Объект Подсети

- **id**: integer — Уникальный идентификатор подсети.
- **ip**: string — IP-адрес или CIDR-нотация подсети.

Пример:

```
{
  "id": 1,
  "ip": "10.7.24.1"
}
```

Объект Задачи

- **id**: integer — Уникальный идентификатор задачи.
- **ip_addr**: string — IP-адрес целевого устройства.
- **password**: string — Пароль для доступа к устройству (обрабатывайте данные безопасно).
- **power_state**: integer — Желаемое состояние питания (0 для выключения, 1 для включения).
- **start_time**: integer — Временная метка Unix начала задачи.
- **end_time**: integer — Временная метка Unix окончания задачи.
- **status**: integer — Текущий статус задачи (например, 0 для ожидания).
- **timestamp**: integer — Временная метка Unix создания или обновления задачи.

Пример:

```
{
  "id": 1,
  "ip_addr": "192.168.1.10",
  "status": 0,
  "power_state": 1,
  "start_time": 1625068800,
  "end_time": 1625072400,
  "timestamp": 1731601221
}
```

Примеры Команд

Включение Устройств из Сна

Команда:

```
devices/power/on
```

Пример Запроса для Одного Устройства:

```
{
  "ip_addr": "10.0.0.1",
  "execution_time": 1730124169,
  "password": "admin"
}
```

Пример Запроса для Подсети:

```
{
  "ip_addr": "10.0.0.1/24",
  "execution_time": 1730124169,
  "password": "admin"
}
```

Пример Массового Добавления:

```
[
  {
    "ip_addr": "10.0.0.1",
    "execution_time": 1730124169,
    "password": "admin"
  },
  {
    "ip_addr": "10.0.0.1/24",
    "execution_time": 1730124169,
    "password": "admin"
  }
]
```

Выключение Устройств и Создание Расписания

Команда:

devices/power/off

Пример Запроса для Создания Расписания:

```
{
  "ip_addr": "192.168.1.1/24",
  "password": "admin",
  "power_state": 1,
  "start_time": 1704067200,
  "end_time": 1704153600
}
```

Пример Создания Графика:

```
[
  {
    "ip_addr": "10.2.96.0/24",
    "password": "admin",
    "power_state": 1,
    "start_time": 1730408400,
    "end_time": 1730421600
  },
  {
    "ip_addr": "10.2.97.0/24",
    "password": "admin",
    "power_state": 1,
    "start_time": 1730408400,
    "end_time": 1730421600
  }
]
```

Примечание: power_state равный 1 означает перевод устройства в состояние работы, а 0 — в сон.

Обработка Ошибок и Диагностика

Umhost Agent записывает логи ошибок и стандартного вывода, которые могут быть полезны для диагностики проблем.

Просмотр Журналов

Журнал ошибок:

```
sudo tail -f /var/log/umhost.err.log
```

Журнал стандартного вывода:

```
sudo tail -f /var/log/umhost.out.log
```

Распространенные Ошибки

1. **Не удается подключиться к MQTT-брокеру:**
 - Проверьте правильность настроек MQTT в конфигурационном файле.
 - Убедитесь, что брокер доступен и порт открыт.
2. **Агент не запускается:**
 - Проверьте права доступа к исполняемому файлу.
 - Проверьте конфигурационный файл Supervisor на наличие ошибок.
3. **Ошибки при работе с API:**
 - Убедитесь, что вы используете правильные эндпоинты и форматы запросов.
 - Проверьте тело запросов на корректность данных.

Решение Проблем

Перезапуск агента:

```
sudo supervisorctl restart umhost
```

Проверка статуса агента:

```
sudo supervisorctl status umhost
```

Обновление конфигурации и перезапуск:

```
sudo supervisorctl reread  
sudo supervisorctl update  
sudo supervisorctl restart umhost
```

Обновление Umhost Agent

Регулярное обновление Umhost Agent и его зависимостей обеспечивает стабильность и безопасность системы.

Проверка Обновлений

Проверьте наличие новой версии Umhost Agent на официальном сайте или репозитории.

Скачайте новую версию:

```
sudo cp путь_до_новой_версии/umhost-linux-arm64 /etc/umhost/
```

Дайте исполняемые права:

```
sudo chmod +x /etc/umhost/umhost-linux-arm64
```

Перезапустите агент:

```
sudo supervisorctl restart umhost
```

Примечание: Рекомендуется создать резервную копию конфигурационных файлов перед обновлением.

Безопасность

Обеспечение безопасности при использовании Umhost Agent крайне важно для защиты оборудования и данных.

Рекомендации по Безопасности

- 1. Используйте Защищенные Каналы Связи:**
 - Всегда используйте HTTPS для API-запросов.
 - Используйте TLS для подключения к MQTT-брокеру.
- 2. Надежная Аутентификация:**
 - Используйте сложные и уникальные пароли для доступа к устройствам.
 - Регулярно обновляйте пароли.
- 3. Хранение Паролей:**
 - Не храните пароли в открытом виде.
 - Используйте шифрование для хранения и передачи паролей.
- 4. Ограничение Доступа:**
 - Ограничьте доступ к API и MQTT-серверу только доверенным IP-адресам.
 - Используйте брандмауэры и другие средства защиты сети.
- 5. Регулярные Обновления:**

- Обновляйте Umhost Agent и все зависимые компоненты для устранения известных уязвимостей.

6. Мониторинг и Аудит:

- Регулярно проверяйте журналы для обнаружения подозрительной активности.
- Внедрите системы оповещения для своевременного реагирования на инциденты безопасности.