

*MikroTik*

+



Telegram

Евгений Коржиков  
Минск, МУМ 2018

# Обо мне

Евгений Коржиков

Опыт работы с mikrotik более 3 лет

Сертификаты:

**Mikrotik:** МТСНА, МТСРЕ, МТСВЕ, МТСТСЕ

**Asterisk:** dCAP

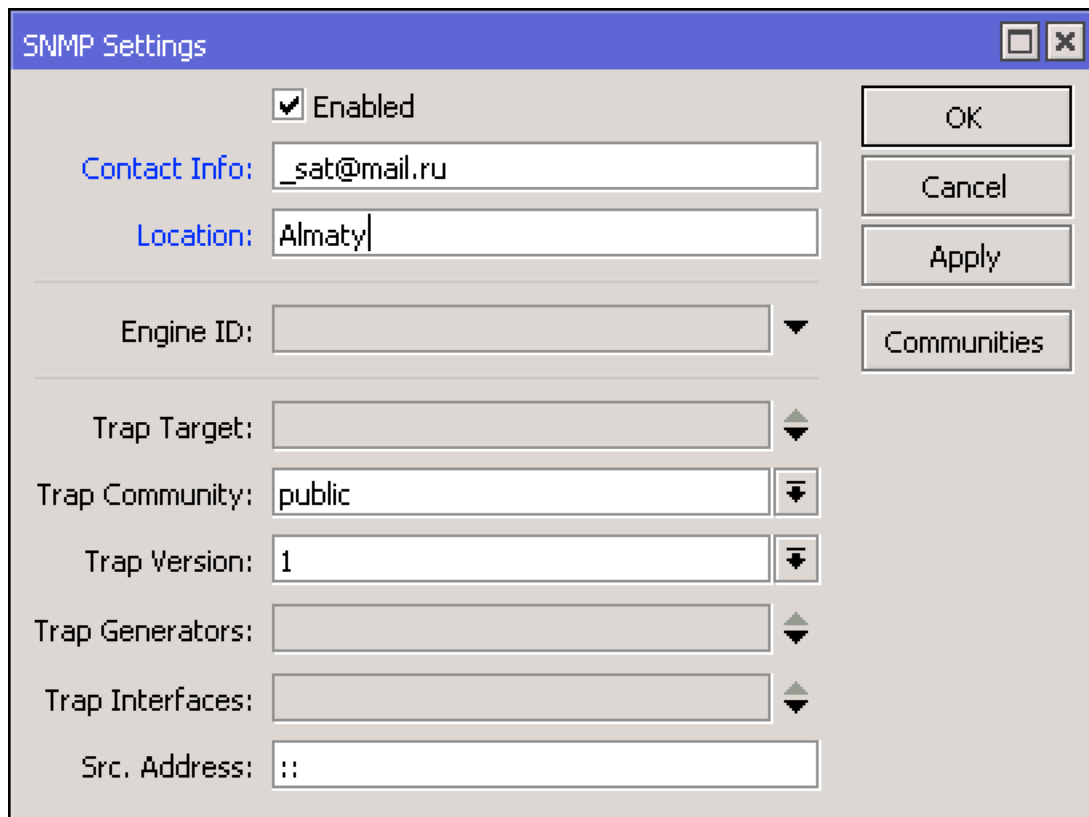
# Цель презентации

Продемонстрировать способ управления устройствами RouterOS посредством мессенджера telegram.

# SNMP

- Протокол для управления и мониторинга устройствами в IP-сетях;
- Поддерживается подавляющим большинством производителей сетевого оборудования;
- Поддерживаемые устройства: маршрутизаторы, серверы, принтеры, различные измерительные приборы;
- Работает через oid коды;
- Можно запускать скрипты на mikrotik устройстве;

# Настройка SNMP



SNMP Settings

Enabled

Contact Info:

Location:

Engine ID:

Trap Target:

Trap Community:

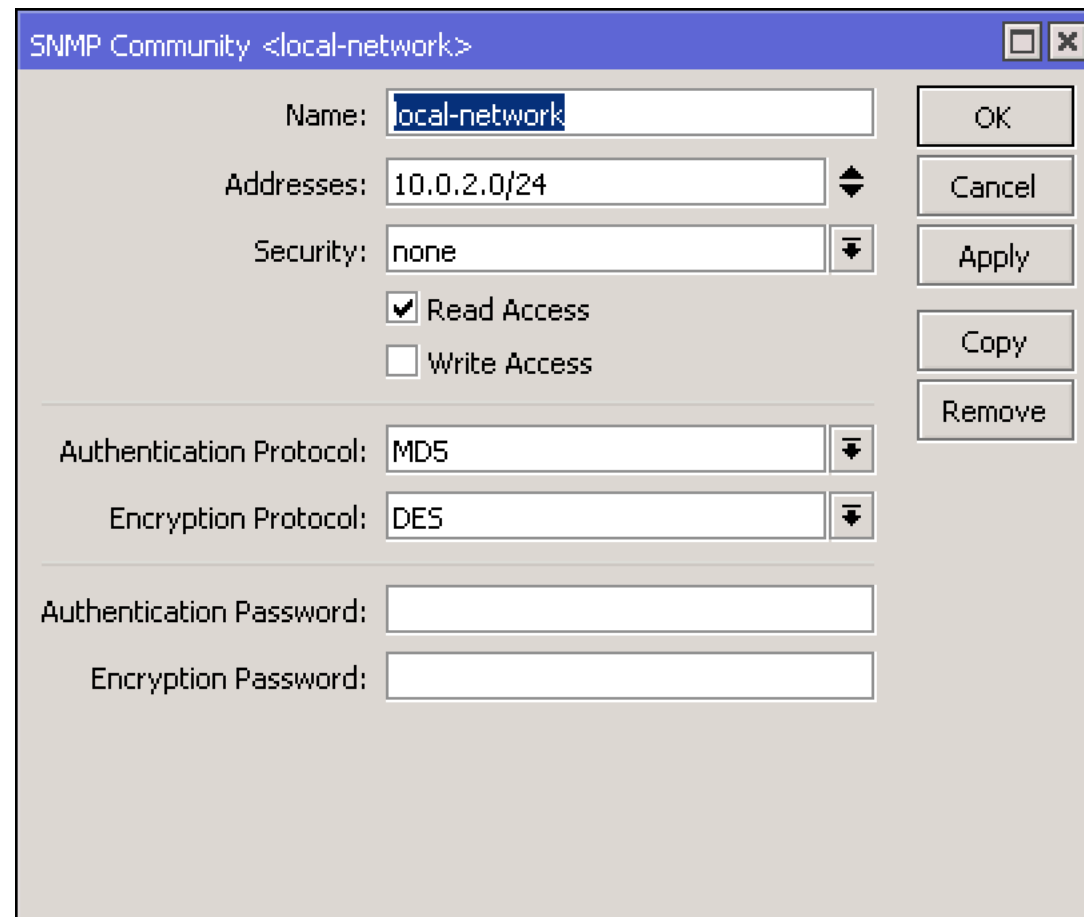
Trap Version:

Trap Generators:

Trap Interfaces:

Src. Address:

OK  
Cancel  
Apply  
Communities



SNMP Community <local-network>

Name:

Addresses:

Security:

Read Access  
 Write Access

Authentication Protocol:

Encryption Protocol:

Authentication Password:

Encryption Password:

OK  
Cancel  
Apply  
Copy  
Remove

# SNMP oid значения

```
[admin@MikroTik Almaty] /system health> print oid
  active-fan: .1.3.6.1.4.1.14988.1.1.3.9.0
    voltage: .1.3.6.1.4.1.14988.1.1.3.8.0
      current: .1.3.6.1.4.1.14988.1.1.3.13.0
        temperature: .1.3.6.1.4.1.14988.1.1.3.10.0
          cpu-temperature: .1.3.6.1.4.1.14988.1.1.3.11.0
            power-consumption: .1.3.6.1.4.1.14988.1.1.3.12.0
```

```
[admin@MikroTik Almaty] /system resource> print oid
  used-memory: .1.3.6.1.2.1.25.2.3.1.6.65536
    uptime: .1.3.6.1.2.1.1.3.0
      build-time: .1.3.6.1.4.1.14988.1.1.7.6.0
        total-memory: .1.3.6.1.2.1.25.2.3.1.5.65536
          cpu-frequency: .1.3.6.1.4.1.14988.1.1.3.14.0
```

```
[admin@MikroTik Almaty] /interface> print oid
Flags: D - dynamic, X - disabled, R - running, S - slave
0 R name=.1.3.6.1.2.1.2.2.1.2.2 actual-mtu=.1.3.6.1.2.1.2.2.1.4.2 mac-address=.1.3.6.1.2.1.2.2.1.6.2
  admin-status=.1.3.6.1.2.1.2.2.1.7.2 oper-status=.1.3.6.1.2.1.2.2.1.8.2 bytes-in=.1.3.6.1.2.1.31.1.1.1.6.2
  packets-in=.1.3.6.1.2.1.31.1.1.1.7.2 discards-in=.1.3.6.1.2.1.2.2.1.13.2 errors-in=.1.3.6.1.2.1.2.2.1.14.2
  bytes-out=.1.3.6.1.2.1.31.1.1.1.10.2 packets-out=.1.3.6.1.2.1.31.1.1.1.11.2 discards-out=.1.3.6.1.2.1.2.2.1.19.2
  errors-out=.1.3.6.1.2.1.2.2.1.20.2
```

# API

- Позволяет создавать собственные программные решения для управления RouterOS устройством;
- Позволяет использоваться интерфейс командной строки (CLI);
- Работает на порту №8728, по умолчанию выключен;

Name	Port	Available From	Certificate
api	8728		
api-ssl	8729		none
ftp	21		
ssh	22		
telnet	23		
winbox	8291		
www	80		
www-ssl	443		none

Name:

Port:

Available From:

OK

Cancel

Apply

Disable

enabled

# Используемые инструменты разработки

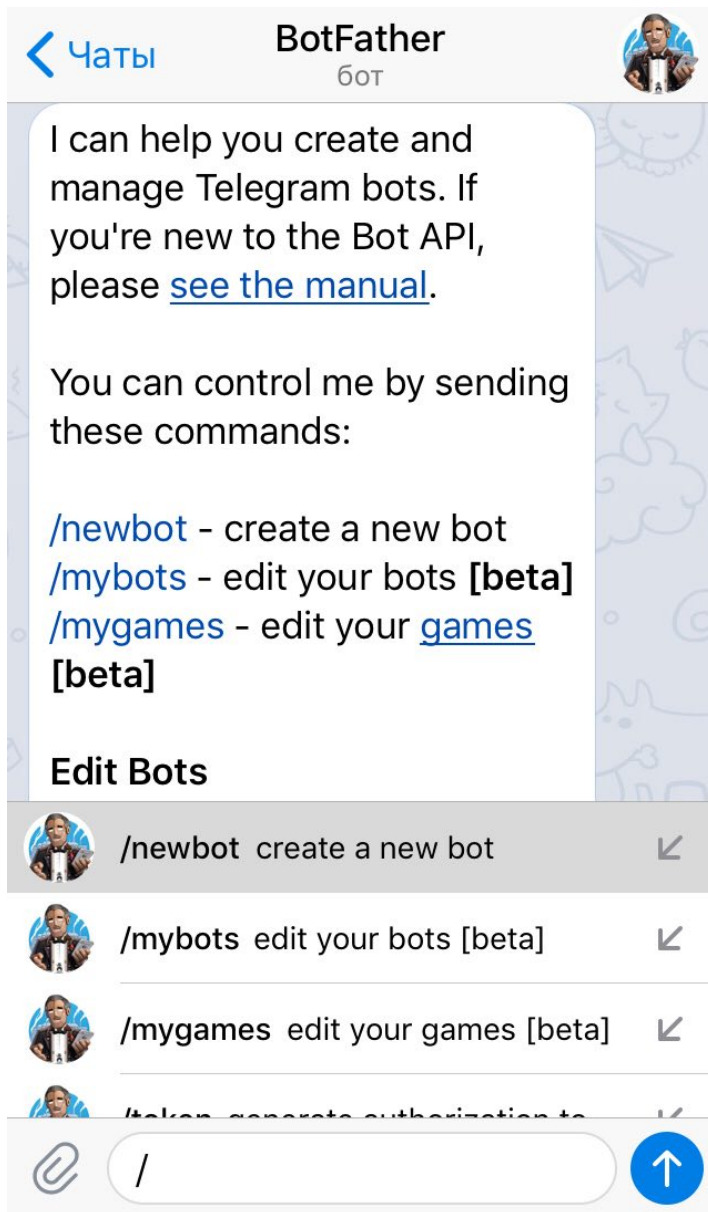
*MikroTik*  
API, SNMP



Telegram Bot API







# Создание бота

- Находим «BotFather» в контактах Telegram;
- Пишем или выбираем, из commands, команду «/newbot»
- Задаем имя боту;
- Получаем Token созданного бота;

```
const TelegramBot = require('node-telegram-bot-api');  
const token = '000000000:XXXxXXxxx0x00xXX0XxXXXXxxX0XXXx0Xx';  
const bot = new TelegramBot(token, { polling: true });
```

# Подключение к API mikrotik Mikronode

```
1 const MikroNode = require('mikronode');
2 var config = require('config');
3 var device_list = config.get('devices');
4 var devices = Array();
5
6 for (var device in device_list) {
7     if (device_list.hasOwnProperty(device)) {
8         device = device_list[device];
9         device.device = new MikroNode( device.ip );
10        device.query = function(query, params, callback){
11            .....this.device.connect().then(([login])=>login(this.login, this.pass)).then(function(conn) {
12                conn.closeOnDone(true);
13                var channel = conn.openChannel("channel");
14                .....channel.write( query, params)
15                channel.done.subscribe(function(packet) {
16                    var results = Array();
17                    packet.data.forEach(function(data1){
18                        var parsed=MikroNode.resultsToObj(data1);
19                        results.push(parsed);
20                    });
21                    callback(results);
22                    chan.close();
23                    conn.close();
24                });
25            });
26        };
27        devices.push(device);
28    }
29 }
```

```
1 {
2     "devices": {
3         "router_name": {
4             "name": "Router_name",
5             "ip": "192.168.88.1",
6             "login": "admin",
7             "pass": "password",
8             "health": {
9                 "voltage": [22,25], //min,max
10                "temperature": [20,40] //min,max
11            }
12        }
13    }
14 }
```

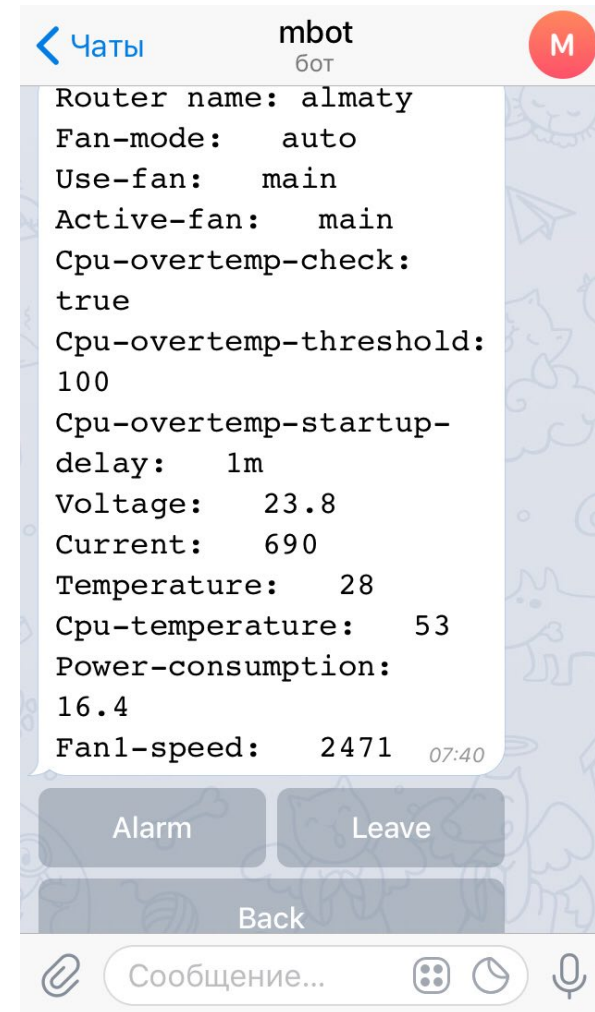
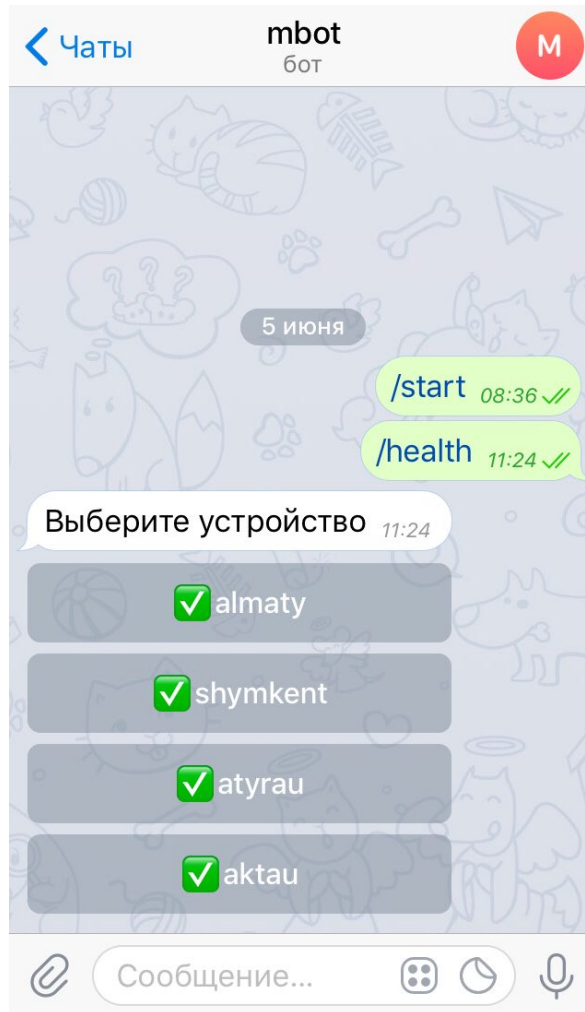
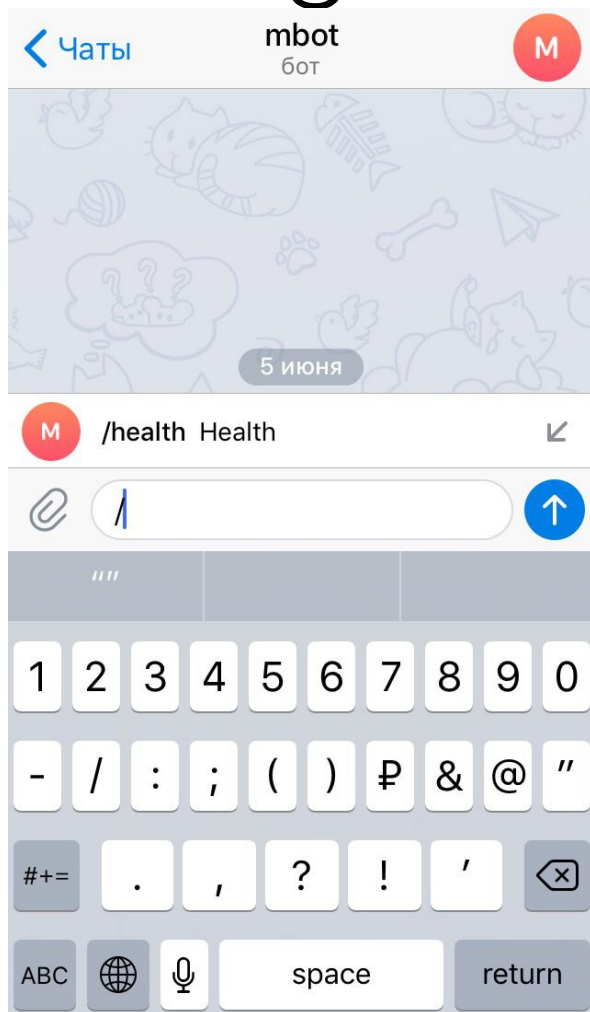
# Подключение к SNMP mikrotik

```
var snmp = require('snmp-native');
var session = new snmp.Session({ host: '192.168.88.1', port: 161, community: 'public' });
session.get({ oid: [1,3,6,1,4,1,14988,1,1,3,10,0] }, function (error, varbinds) {
  if (error) {
    console.log('Fail :(');
  } else {
    console.log(varbinds);
    console.log(varbinds[0].value);
  }
}
```

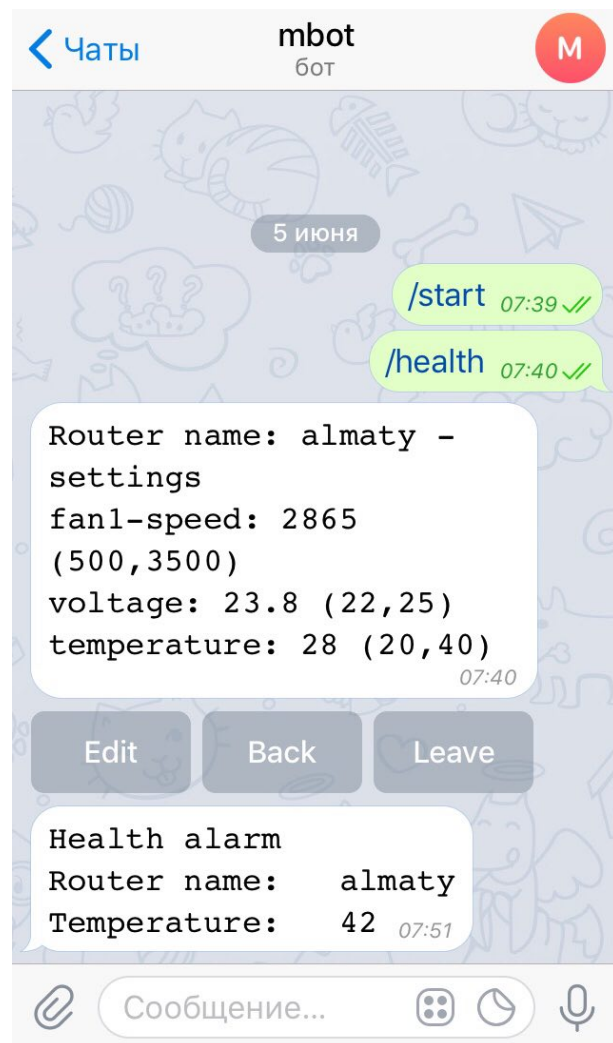
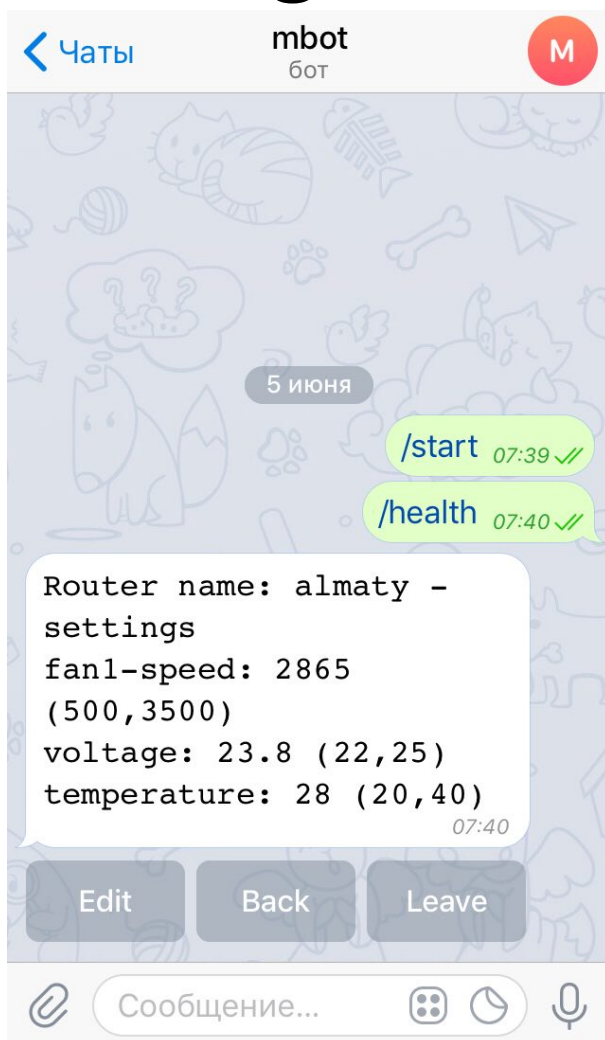
# NodeJS и API mikrotik

```
devs[i].query('/system/health/print',Array(),function(data) { /*code*/ });  
devs[i].query('/queue/simple/print',{['?=target=192.168.88.254/32']:''},function(data) { /*code*/ });  
devs[i].query('/queue/simple/set', ['=max-limit=100000/100000', '=id=*2BE'],function(data) { /*code*/ });  
devs[i].query('/queue/simple/add', ['=max-limit=100000/100000', '=target=192.168.88.10/32'],function(data) { /*code*/ });  
devs[i].query('/ip/dhcp-server/lease/print',{['?=.id=*2BE']:''},function(data) { /*code*/ });  
devs[i].query('/ping',{address:S('192.168.88.1').s, |count:3},function(data) { /*code*/ });
```

# Telegram Bot Health



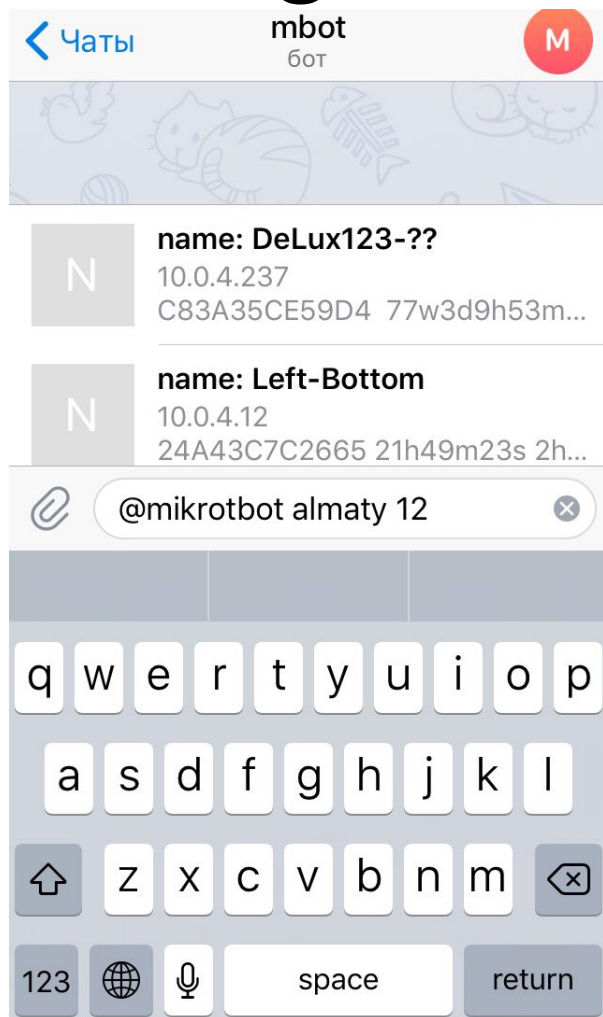
# Telegram Bot Health



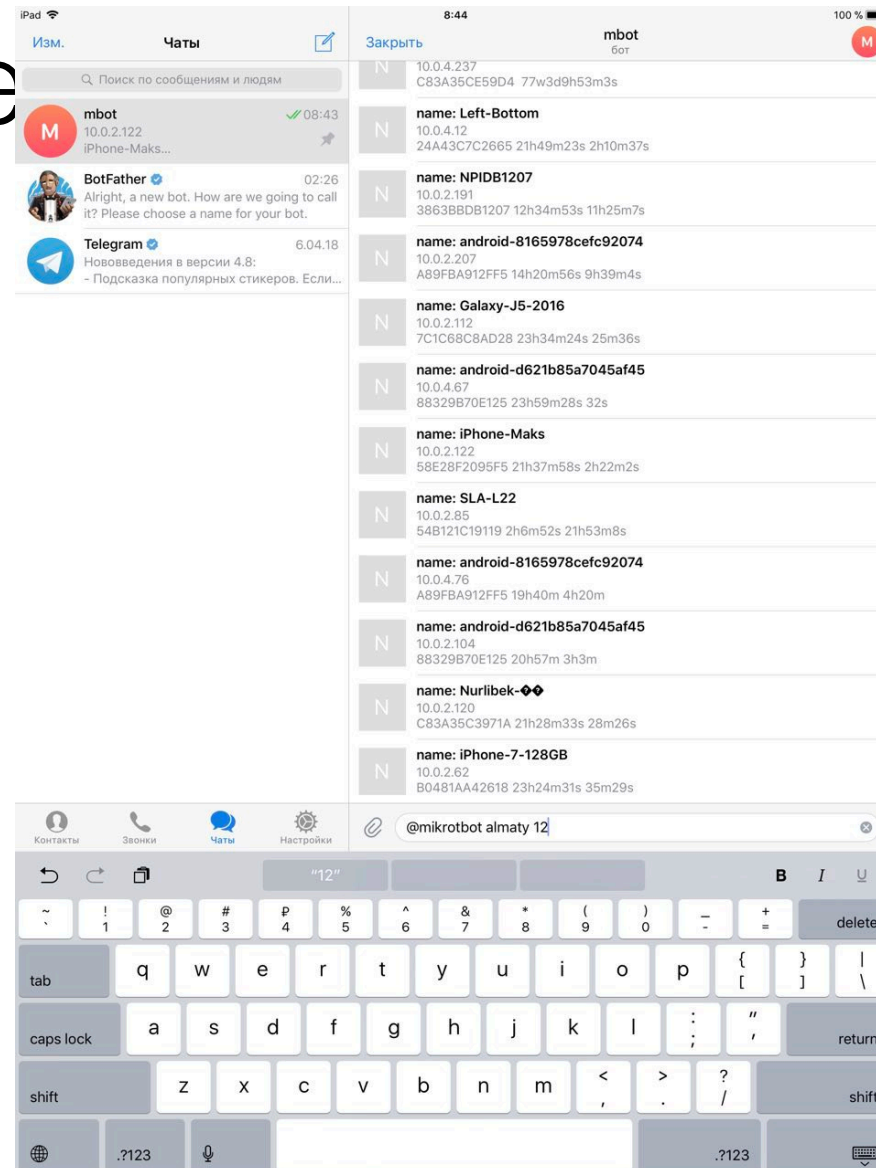
- Настраиваем параметры сигнализации;
- Бот сообщит в виде сообщения, если показатели выйдут за границы;



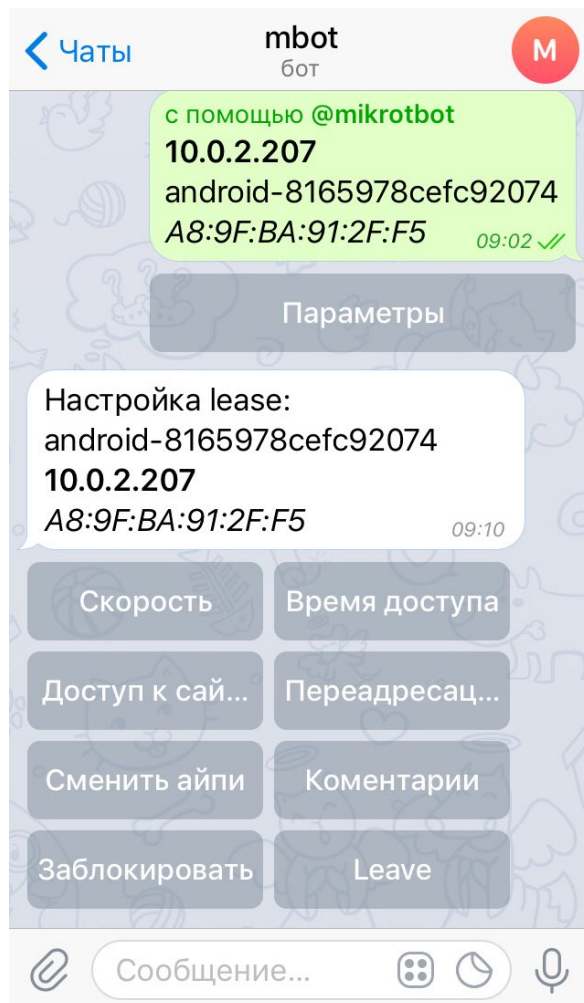
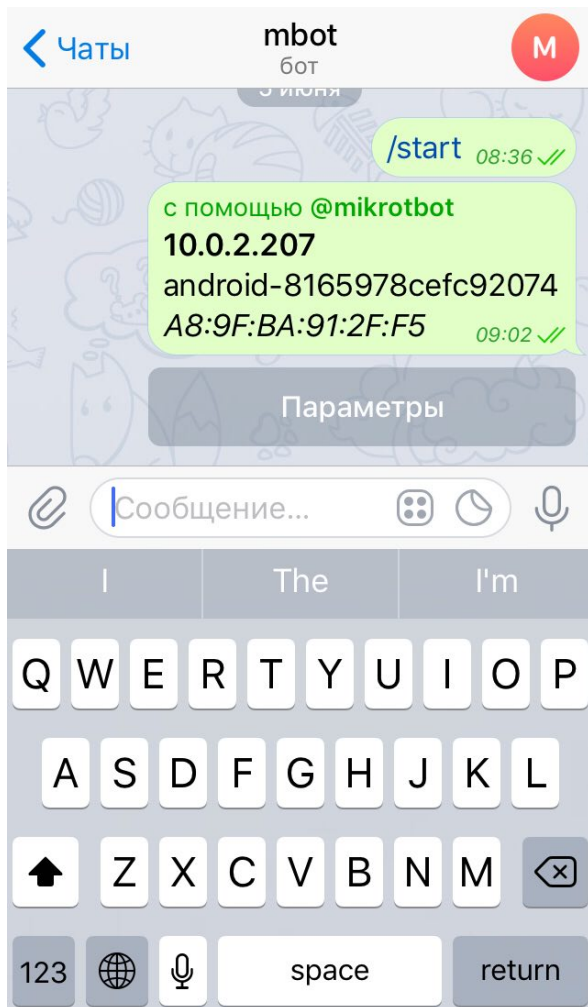
# Telegram Bot DHCP Lease



- Пишем имя бота через символ @, т.е. Вызываем бота через inline query;
- Далее передаем боту 2 параметра, первый - имя устройства, второй текст для поиска;
- Бот ищет по списку DHCP leases, по полям: ip, mac, host-name;
- Не более 50 результатов



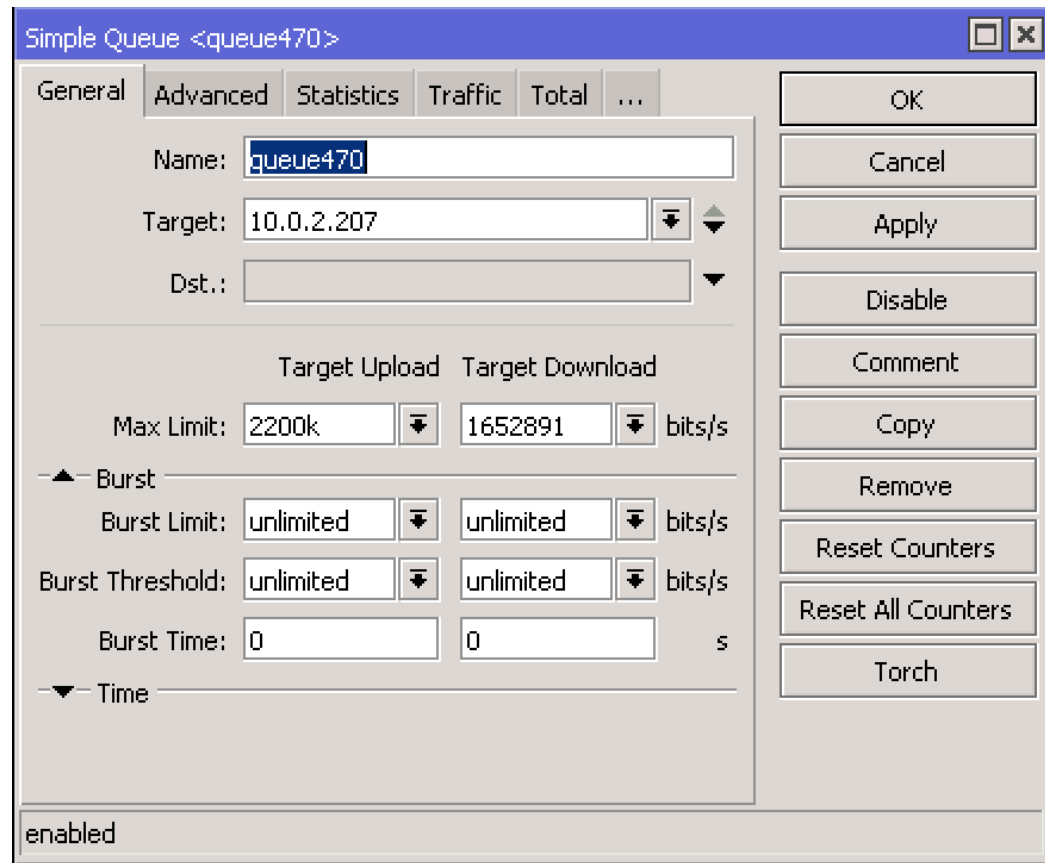
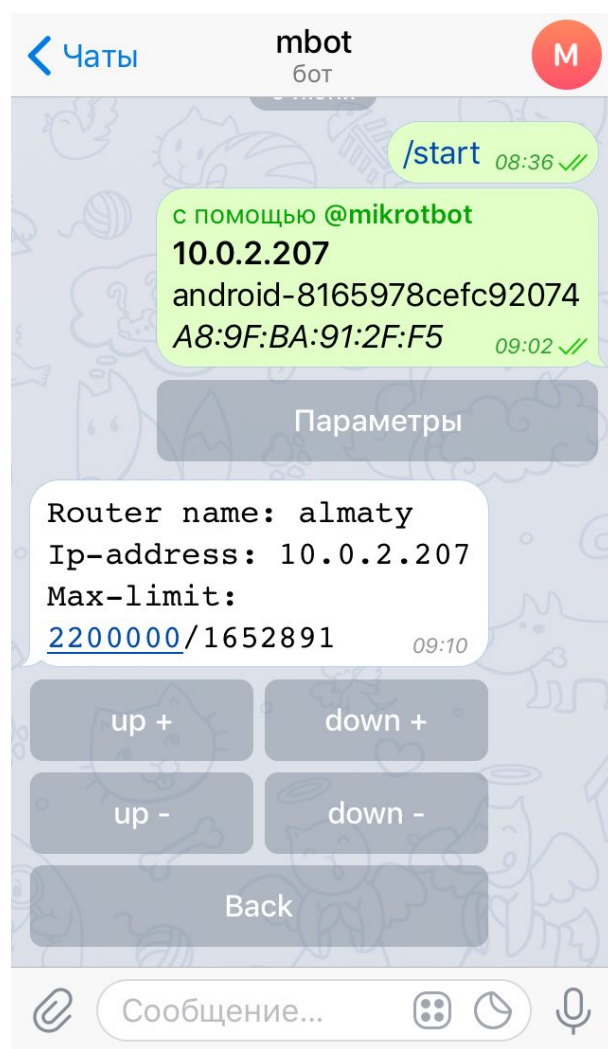
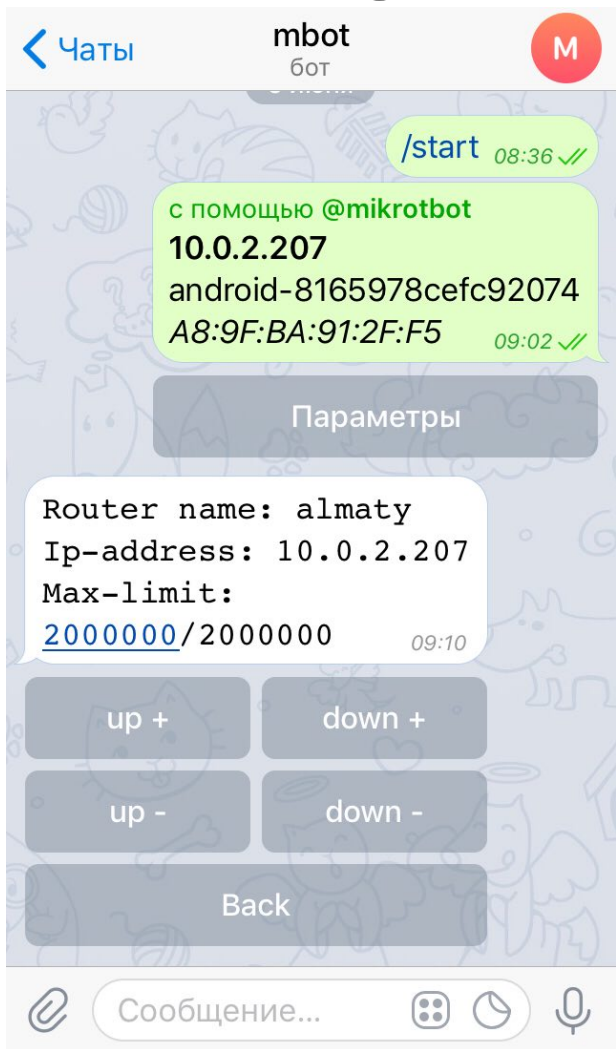
# Telegram Bot DHCP Leases



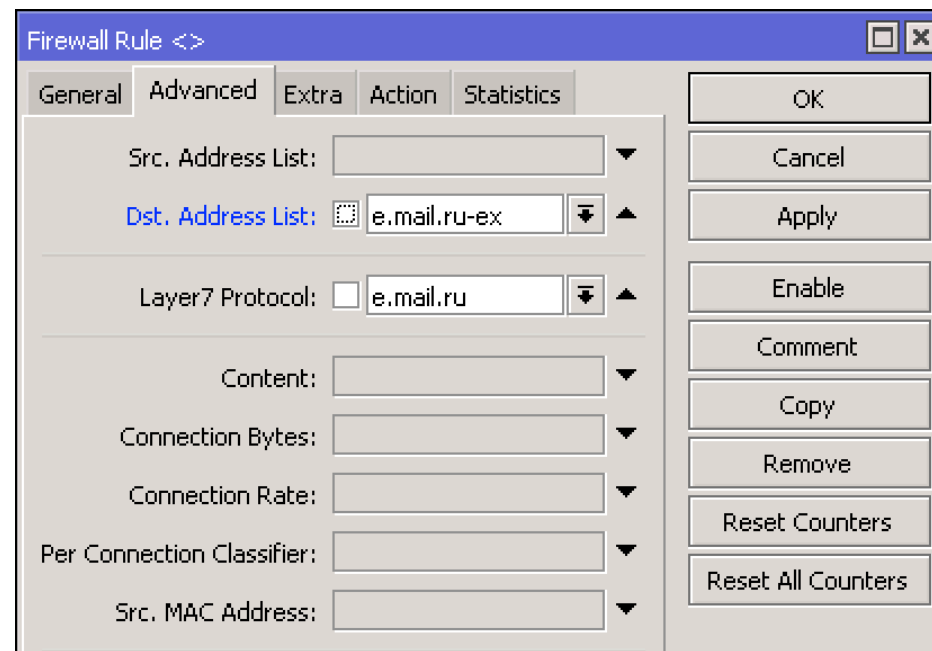
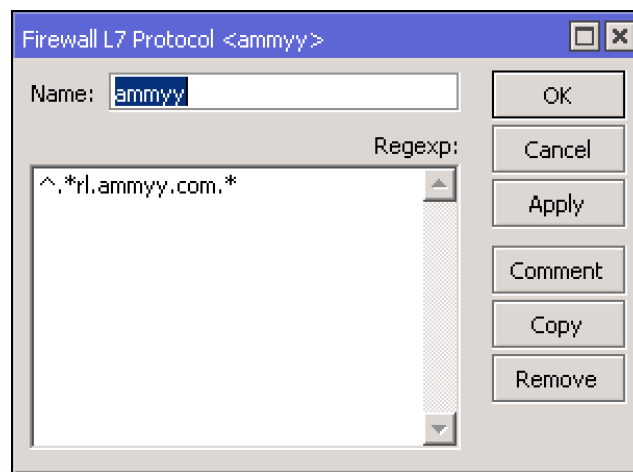
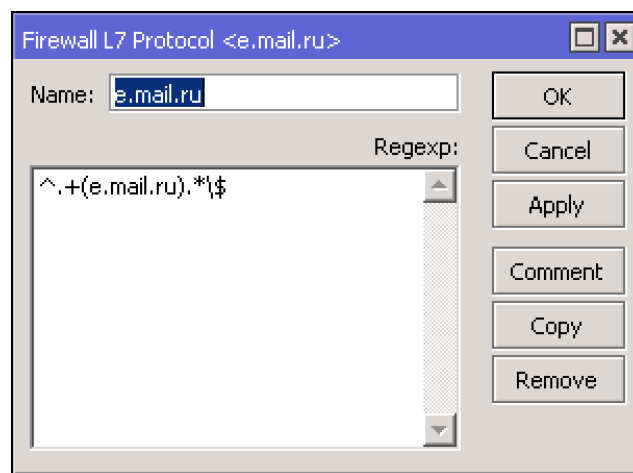
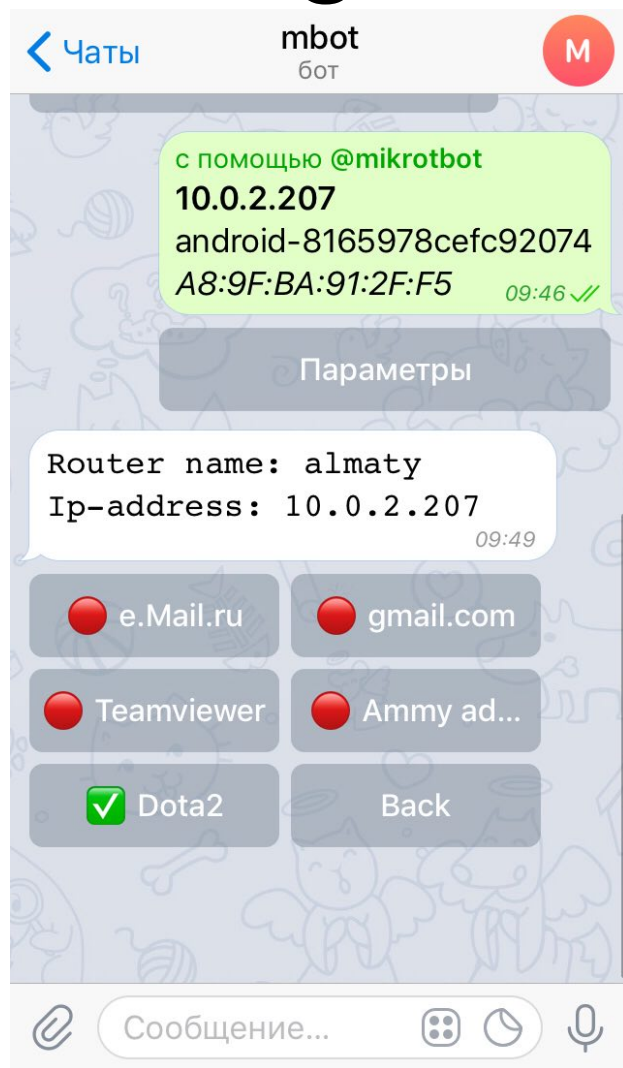
- После выбора lease, настраиваем его, посредством заложенных функций;
- Скорость - simple queue;
- Время доступа - simple queue;
- Доступ к сайтам - firewall;
- Переадресация - routes;
- Сменить IP- dhcp lease;
- Комментарии - dhcp lease;
- Заблокировать - firewall;



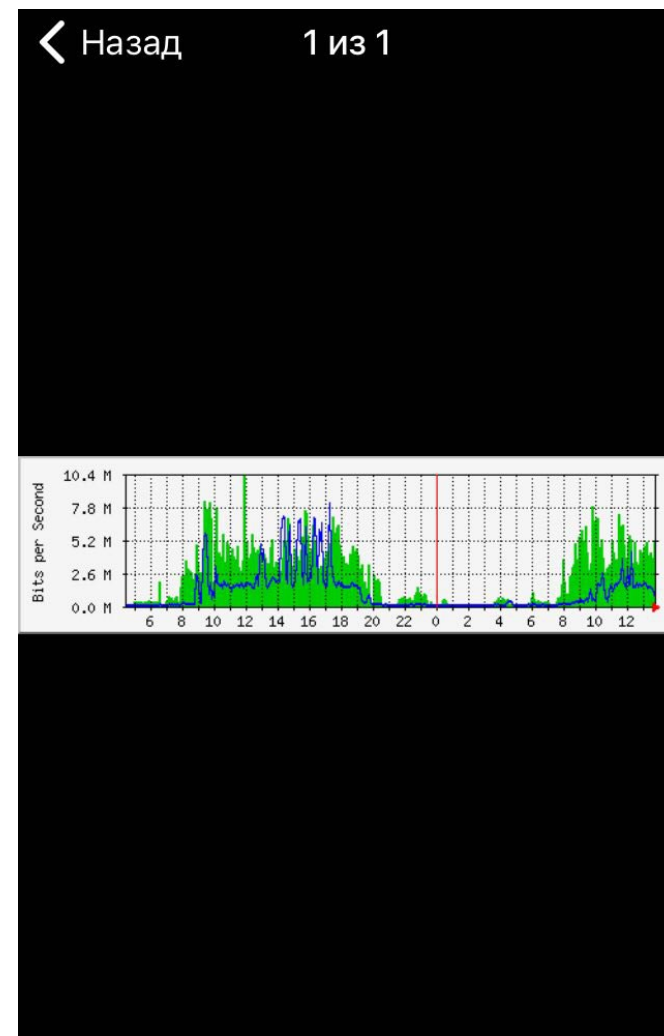
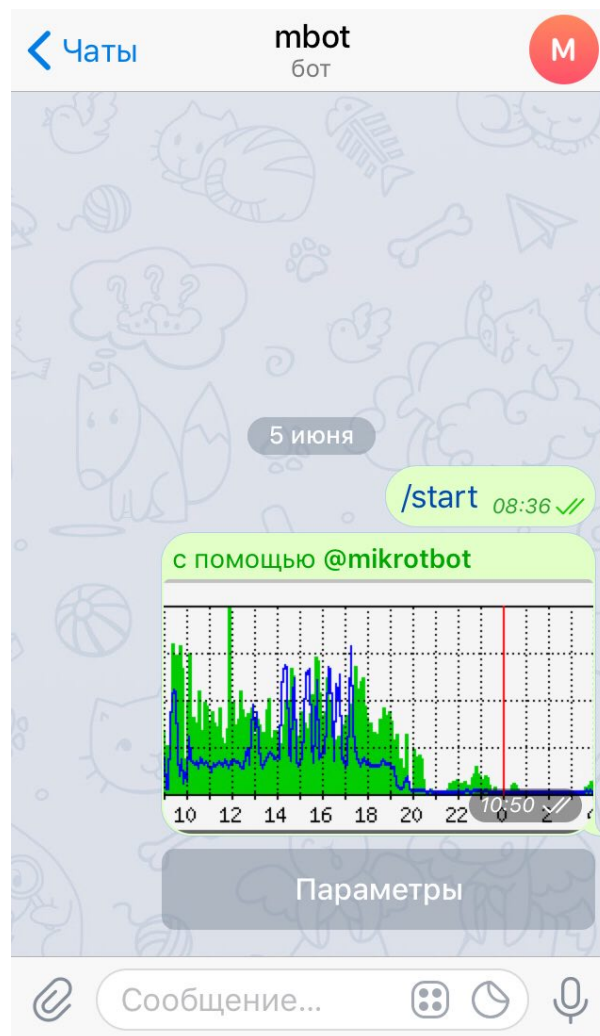
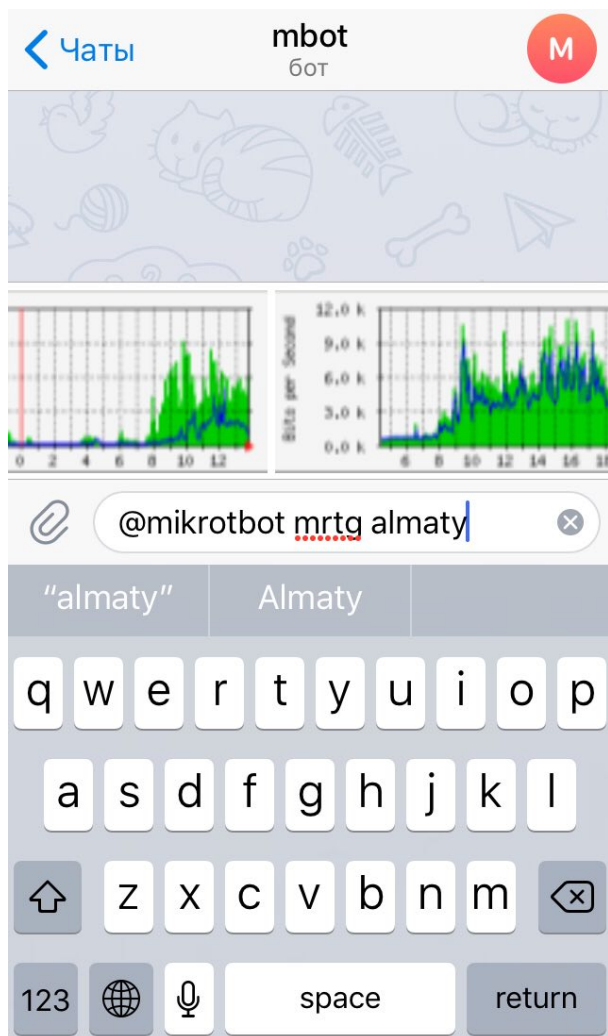
# Telegram Bot DHCP Leases queue



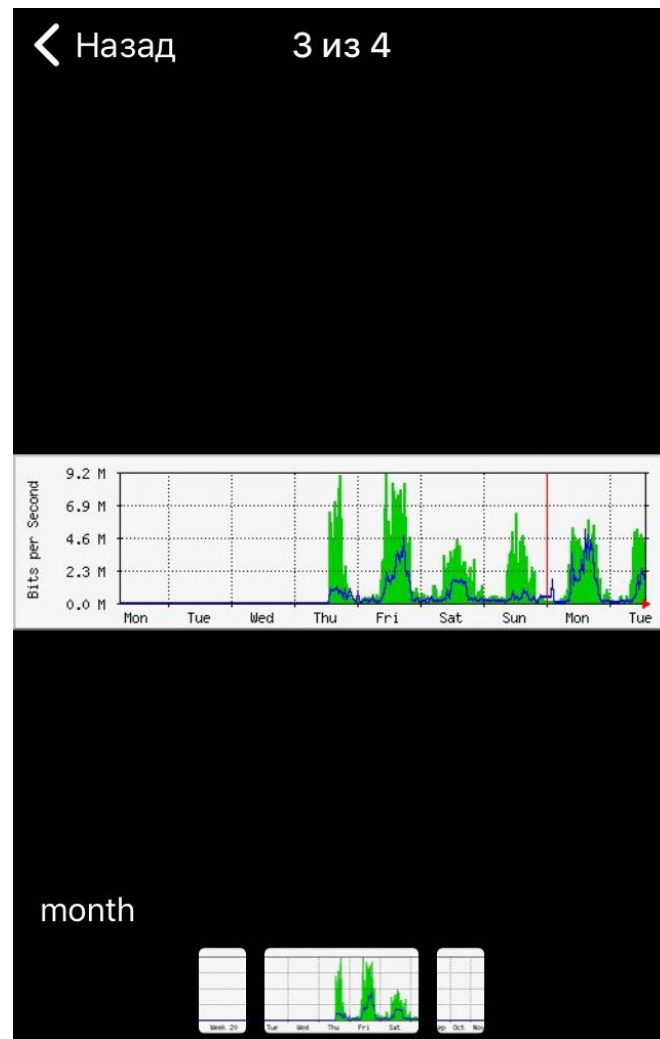
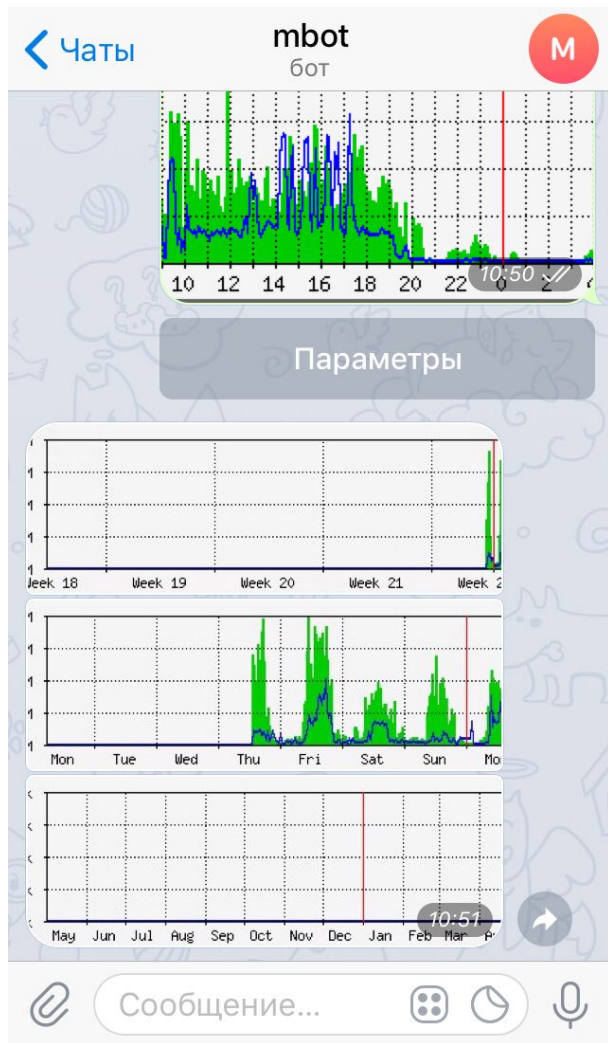
# Telegram Bot DHCP Leases firewall



# Telegram Bot MRTG



# Telegram Bot MRTG



## Дополнительная информация:

- <https://wiki.mikrotik.com/wiki/Manual:SNMP>
- <https://wiki.mikrotik.com/wiki/Manual:API>
- <https://github.com/Trakkasure/mikronode>
- <https://core.telegram.org/api>
- <https://github.com/yagop/node-telegram-bot-api>
- <https://nodejs.org/en/>
- <https://www.npmjs.com/package/snmp-native>

# Спасибо за Внимание

Евгений Коржиков  
\_sat@mail.ru