

# WebAPI

This is RESTful JSON WebAPI to manage MT4/MT5 servers from any HTTP-enabled client.

For real-time data it has SignalR endpoint offering WebSockets, LongPooling etc protocols.

The project hosted at <https://cloud.myWebAPI.com> (and other URLs, keep reading for more info)

 There is Swagger 2.0 documentation (<https://cloud.mywebapi.com/swagger/index.html>) to let you very easily generate client-side proxy classes. Do have a look at <https://github.com/swagger-api/swagger-codegen> This saves a lot of time when you want to generate proxy classes for very first time. But this is not an obligation. Everything might be done without this step.

## How to use

First of all you have to be registered within system.

If you don't have registration and would like to create new organization item from scratch than you can navigate to any website from the list and pass steps mentioned there:

- <https://admin.cplugin.net> - being logged off it will redirect you to authorization server to authenticate or register brand new account.
- <https://auth.cplugin.net> - here you can sign in or sign up. Then you can navigate to other web sites form family of cloud products.
- etc

Then navigate to <https://admin.cplugin.net> Clients and create api account for your application. Remember login and password, you will not be able to get it later.

When you get client id and secret you can pass authorization step (which explained in another section) and start working with WebAPI (JSON and SignalR endpoints).

### Firewall setting you suggested to set up

IP list from where it can connect to your MT4 servers are:

MS Azure cloud in Western Europe:

13.93.15.230  
13.93.11.147  
13.93.9.186  
13.93.12.49  
52.157.155.68  
52.157.155.75  
52.157.155.82  
52.157.155.86  
13.95.64.33  
13.81.215.235  
13.69.68.50

Mirrors hosted around the world:

116.203.101.91 - Germany, [m2.mywebapi.com](https://m2.mywebapi.com)  
194.233.70.35 - Singapore, [m1.mywebapi.com](https://m1.mywebapi.com)

## Cloud - <https://cloud.myWebAPI.com>

We run a swarm of Web API instances to archive few goals:

- geographical distributed instances for minimize delay in communication between caller and server
- multiple instances must gain up uptime
- spread the load over multiple servers
- if any instance down, you get IP address of another instance automatically. It is a matter of second when systems reacts.

All above made possible by DNS traffic manager, if you try to get IP of domain name `c1oud.mywebapi.com` - you will get IP address of instance, which is closest (geographically) to you (with least ping).

We regularly analyse usage and can run additional server in the region of customer, who can get real benefits of that.