

Bithouse WEB Engine

Web Service API

- [Web service-rajapinta](#)
- [Kirjautuminen](#)
- [Datapisteet](#)
- [Historiadataan lukeminen](#)

Web service-rajapinta

Web service -rajapinnan voidaan kutsua seuraavia operaatioita.

Operaatio	Metodi	URL-pohja	Parametrit
Kirjautuminen (käyttäjä)	POST	plc.php	user=[username] secret=[passwd]
Kirjautuminen (robotti)	GET	plc.php	&key=[keyString]&id=[idString]
Lue pisteen arvo	GET	plc.php	&get=[pointName]
Kirjoita pisteen arvo	GET	plc.php	&set=[pointName]&value=[json]
Historia-datan lukeminen	GET	plc.php	&historybuffer=[bufferName] &db=[historyfile]&time=[starttime]
Listaa pisteet	GET	plc.php	&list=[namefilter]&fields=[fieldfilter]

HUOM! Operaatioita voidaan suorittaa vain onnistuneen kirjautumisen jälkeen.

Rajapinnassa on **kaksi eri kirjautumistapaa**:

- POST-pohjainen kirjautuminen on tarkoitettu ihmisten käytettäväksi käyttöpaneelien kautta. Luo uuden session. (ei siis yleensä web service käyttöön).
- GET-pohjainen (shared key -tyyppinen) kirjautuminen on tarkoitettu roboteille ja automaattisten tiedonkeräysjärjestelmien käytettäväksi (ei luo sessiota).

Jaettu tunnistautumisavain syötetään laitteeseen käyttöönoton yhteydessä. Rajapinnan tarjoama data on JSON-muotoista.

Versiossa 3 voi valvomon kautta suorittaa kyselyjä myös ala-asemille. Tällöin ala-aseman NodeID annetaan "remote" -parametrina, mutta muilta osin komento on sama.

Kirjautuminen

Käyttäjä

Käyttäjän kirjautuminen on tarkoitettu käyttöpäätteille. Käyttäjä kirjataan sisään ja luodaan tarvittaessa uusi sessio palvelimelle asiakasta varten.

Esimerkki:

```
/plc.php
```

```
POST
```

```
user=kayttaja  
secret=1111
```

```
Vastaus:
```

```
[{"time":"1505129923"}]
```

Esimerkki:

```
/plc.php
```

```
POST
```

```
user=kayttaja  
secret=111111
```

```
Vastaus:
```

```
[{"error":true,"code":99,"msg":"Authentication failed"}]
```

HUOM! Ilman sisäänkirjautumista ei voi käyttää mitään muuta web-rajapinnan toimintoa!

Robotti

Robottimallinen kirjautuminen suorittaa sisäänkirjautumisen palvelulle, automaattista tietojen siirtoa varten.

Käyttäjällä joka kirjautuu tällaisella avaintunnuksella (shared key) ei ole täysiä valtuuksia, mm. arvojen kirjoittaminen tietokantaan tai PLC-skriptien suorittaminen web service -rajapinnan yli ei ole sallittua.

Esimerkki:

```
/plc.php?key=12312345
```

Vastaus:

```
[{"time":"1505129923"}]
```

Esimerkki:

```
/plc.php?key=133333335
```

Vastaus:

```
[{"time":"1505129923"}]
```

Vastaus:

```
[{"error":true,"code":99,"msg":"Authentication failed"}]
```

HUOM! Ilman sisäänkirjautumista ei voi käyttää mitään muuta web-rajapinnan toimintoa!

Yhdistetty kirjautuminen ja kysely

Shared key -autentikointiavain on mahdollista antaa suoraan mm. list- tai get-kyselyiden yhteydessä URL-parametrina, jolloin erillistä kirjautumista ei tarvita.

Esimerkki:

```
/plc.php?list=ioPoints/%2A&key=1234512345
```

Vastaus:

```
[{"list":["ioPoints\TK01\TE10","ioPoints\TK01\TE30"]},{"time":"1505137744"}]
```

Datapisteet

Lukeminen

Read datapoint value lukee datapisteiden arvoja WS-rajapinnan ylitse. Samalla komennolla voidaan lukea useita pisteitä erottamalla niiden nimet puolipilkulla (;).

Esimerkki: Luetaan koko piste "sys/settings/networkSettings".
/plc.php?get=sys/settings/networkSettings

Vastaus:

```
[{"get": "{\"defaultGateway\": \"192.168.0.1\", \"subnetMask\": \"255.255.255.0\", \"ipAddress\": \"192.168.0.128\", \"dnsServer\": \"8.8.8.8\", \"broadcastAddress\": \"192.168.0.255\"}", {"time": "1505132981"}]
```

Esimerkki: Luetaan kahden prosessidatapisteiden oloarvot (pisteet ioPoints/TK01/TE10 ja ioPoints/TK01/TE30).

/plc.php?get=ioPoints/TK01/TE10.pv;ioPoints/TK01/TE30.pv

Vastaus:

```
[{"get": "20.6"}, {"get": "19.6"}, {"time": "1505133163"}]
```

Esimerkki: Luetaan sellaisen pisteen arvo, jota ei ole olemassa.

/plc.php?get=ioPoints/TK01/TE100.pv

Vastaus:

```
[{"get": "-4294967296"}, {"time": "1505133224"}]
```

kirjoittaminen

Datapisteiden arvon kirjoittaminen WS-rajapinnan ylitse.

Value-parametri, eli kirjoitettava arvo annetaan JSON-muodossa.

Huom! Ei käytössä shared key -autentikoinnin kautta.

Esimerkki: Kirjoitetaan tuloilman asetusarvo (piste "ioPoints/TK01/TE10_SP").

```
/plc.php?set=ioPoints/TK01/TE10.pv&value=22.0
```

Vastaus:

```
[{"set":"true"}, {"time":"1505137224"}]
```

Listaus

Proessorin datapisteiden listaaminen. Suodatusta listassa voidaan tehdä joko pistenimissä (wildcard-merkillä *) tai niiden sisältämien arvojen perusteella.

Syntaksi *list*- ja *field*-parametreille on sama kuin Data.list() -kutsussa.

list Tällä parametrilla annetaan suodausehto pisteiden nimelle. Wildcardina toimii merkki '*', joka on URL-koodattuna %2A.

field Tämän parametrin avulla voidaan suodattaa pisteitä kenttien ja jopa kenttien arvojen perusteella. Tässä yhteydessä ei voi suoraan käyttää Wild card -merkkejä, koska lausekkeen täytyy olla validi vertailu.

Esimerkki 1: Listataan kaikki TK01:n pisteet, eli nimisuodattimena "ioPoints/TK01/*".

```
/plc.php?list=ioPoints/TK01/%2A
```

Vastaus:

```
[{"list":["ioPoints\TK01\TE10","ioPoints\TK01\TE30"]}, {"time":"1505137744"}]
```

Esimerkki 2: Listataan kaikki pisteet joilla on kenttä 'dataSource'

```
/plc.php?list=ioPoints/TK01/%2A&field=(dataSource)
```

Esimerkki 3: Listataan kaikki pisteet joilla kenttä pv on suurempi kuin 9.0

Muista että field parametrille täytyy tehdä URL enkoodaus! Selväkielisenä vertailu on (pv > 9.0)

/plc.php?list=ioPoints/TK01/%2A&field=(pv%20%3E%209.0)

Historiadataan lukeminen

Actiweb tallentaa historiadataan (mm. trendi-data) time series -tietokantaan. Tätä dataa on mahdollista kysellä "historybuffer"-operaatiolla.

Parametreina pitää antaa tietokantatiedoston nimi ja luettavan datapuskurin nimi. Datapuskurin nimi on aina sama kuin logattavan pisteen nimi yhdistettynä seurattavan parametrin nimeen - yleensä pisteestä seurataan **pv**-parametriä, eli mittauksen oloarvoa - t.s. lukemaa. Muita parametreja ovat mm. **description** joka tarkoittaa selitetekstiä, **dispUnit** eli yksikkö. Historiapuskuriin tallennetaan kuitenkin normaalisti vain mittauksen pv kenttä. Näin puskurin nimeksi muodostuu esimerkiksi "ioPoints/TK01/TE10.pv".

Samaa pistettä voidaan tallentaa useaan eri tiedostoon. Jokaisella historia-tiedostolla on omat erikseen määritellyt näytteenottovälinsä, ja maksimikokonsa, jonka jälkeen vanhimpia näytteitä aletaan poistaa.

Historiatiedostoja ja puskuireita voidaan luoda, poistaa ja asetuksia säätää web-käyttöliittymän ylitse.

Ohjelmisto luo automaattisesti "quicktrend"-nimisen historiatiedoston, jonka näytteistysväli on (oletuksena) 30 sekuntia, ja koko 3000 näytettä. Tähän tiedostoon luodaan datapuskuri kaikille niille tietokannasta löytyville pisteille, joilla on "pv"-parametri (eli nykyinen oloarvo).

GET-operaation parametrilla **time** voidaan määrittää aika, jonka jälkeinen data puskurista halutaan lukea. Aika annetaan n.s. unix epookkina, eli sekunteina ajanhetken 1970.1.1 klo 00:00 jälkeen.

Aika voidaan antaa myös negatiivisena lukuna, jolloin sen tulkitaan tarkoittavan sekuntimäärää ennen nykyhetkeä. Toisinsanoen &time=-86400 tarkoittaa, että halutaan lukea näytteet, joiden aikaleima on alle 24 tuntia vanha.

Esimerkki: Luetaan pikatrendi (tiedosto **quicktrend**) pisteeltä "ioPoints/TK01/TE10" ja parametrilta "pv", edellisen 5 minuutin ajalta.

```
/plc.php?db=quicktrend&historybuffer=ioPoints%2ATK01%2ATE10.pv&time=-300
```

Vastaus:

```
[{"historybuffer":[{"1505743059","1.000000"},{"1505743089","21.100000"},{"1505743119","21.300000"},{"1505743149","21.200000"},{"1505743179","21.300000"},{"1505743209","21.300000"},{"1505743239","21.400000"},{"1505743269","21.300000"},{"1505743299","21.200000"},{"1505743329","21.200000"}]},{time:"1505743354"}]
```


Esimerkiksi kulutusmittaukset tallennetaan tavallisesti **reports** nimiseen tietokantaan, johon oletuksena tallennetaan lukema 30 min välein kellon aikaan sidotusti niin, että näyte tallennetaan tasatunnein, ja puolelta. Alla olevalla kutsulla saa luettua viimeisen 32 päivän lukemat **reports** puskurista, ja mittarilta **mittaukset/HUONEISTO_A10/VM01**.

/plc.php?db=reports&historybuffer=mittaukset/HUONEISTO_A10/VM01.pv&time=-2764800

Vastaus:

```
[{"historybuffer":[{"1505743059","1.000000"},{"1505743089","21.100000"},{"1505743119","21.300000"},{"1505743149","21.200000"},{"1505743179","21.300000"},{"1505743209","21.300000"},{"1505743239","21.400000"},{"1505743269","21.300000"},{"1505743299","21.200000"},{"1505743329","21.200000"}]},{ "time":"1505743354"}]
```