Application Programming Interface (API) BeeClear energy manager
Version 0.0.2

Layout:
Explanation : This is the text explaining the function of this API command.
Command : **bc_command**
Options : Option1 | Option2 | Option3
Options Explained :
Option1 This is the first option it is a decimal number that represents first option
Option2 This is the second option it is a decimal number that represents second option
Option3 This is the third option it is a decimal number that represents third option
Example : http://{BeeClear}/bc_**command**&{**Option1**}=1&{**Option2**}=2&{**Option3**}=3
Returns : Shows examples of returned values with explanation if applicable

1A. Meter values Electricity low/high and Gas (totals)

Explanation : This command retrieves the primary "smart meter" current values (in Dutch: "meterstanden"). The primary meter values are: used normal tariff (ul), used high tariff (uh), generated normal tariff (gl), generated high tariff (gh), current electricity usage (u) and total used gas (gas).

Command : **bc_usage**
Options : **date| duration | period**
Example : http://{BeeClear}/bc_usage?date=1445554800&duration=168&period=24
Returns :

{"d":1429103361,	// d: date unix time	(time of most recent meter telegram data)
"ul":1061049,	// ul: used low (KWH * 1000)	(meter total used low)
"uh":883337,	// uh: used high (KWH * 1000)	(meter total used high)
"gl":123574,	// gl: generated low (KWH * 1000)	(meter total generated low)
"gh":334551,	// gh: generated high (KWH * 1000)	(meter total generated high)
"u":0.000,	// u: current usage in KW	(current active electricity usage)
"g":1.934,	// g: current generating in KW	(current active electricity generating)
"gas":1142889 }	//gas: gas usage in cubicle meters m3 * 1000	(meter total used gas)

The smart meter returns every 10s a telegram with actual meter values for both gas and electricity. An additional field indicates active usage.

The Beeclear stores all these values and syncs them every hour to SD-card.

The values are stored in separate DB's

Formats:
1 x per 10 seconds - Every Telegram
1 x per 60 seconds - The average value of 6 Telegrams
1 x per 360 seconds - The average value of 36 Telegrams
1 x per 2160 seconds - The average value of 216 Telegrams

Bc_usage&date... returns a value from 1 of the three DB;s depending on requested length
From which DB data is returns depends on the requesting length. Aribters are based per 1000 boundry.
Depending on period length you can choose which DB you read.

1B. Recorded meter values Electricity low/high and Gas (totals)

Explanation : This command with options retrieves the history of **recorded** meter values from the database and returns a JSON array. The options date, duration and period are responsible for filling an array of recorded historic meter values.

Command : **bc_usage**
Options : **date|duration|period**
Options Explained :
date In (Unix epoch time), set the end date of measurement.
duration in units of 1 hour (integer) the time between two measured points (resolution) default is 1 hour if omitted.
period Set history view period back in time. Starting point is set by **date**. Valid property options are: {day|week|month|year|number}.
Example : http://{BeeClear}/bc_usage?duration=1&period=day&date=1435322813
Returns :

{"items":[{
"d":1428652200, // d: date unix time (time of measurement)
"s":"4530303039313030303038303639373134", // s: serial number (serial number of electricity smart meter)
"ul":873705, // ul: used low (KWH *1000) (meter total used low @d)
"uh":1037095, // uh: used high (KWH *1000) (meter total used high @timestamp d)
"gl":295602, // gl: generated low (KWH *1000) (meter total generated low @d)
"gh":105674}, // gh: generated high (KWH *1000) (meter total generated high @d)
{"d":1428655800,"s":"4530303039313030303038303639373134","ul":873784,"uh":1037095,"gl":295694,"gh":105674},
..... // datasets with measured values
{"d":1428825000,"s":"4530303039313030303038303639373134","ul":876403,"uh":1046996,"gl":304698,"gh":112877}]}

2. Recorded historic active values electricity used and -generated (low/high tariff)

Explanation : This command returns **recorded** historic active values for electricity used and generated for tariffs high and low. The returned data has a per 10 sec interval resolution.

Command : **bc_current**
Options : **date|duration|nu**
Options Explained :
date (unix epoch time), set the ending date/time of measurement.
duration Interval presentation 15 min units. For example 4 units = 1 hour. Resolution is 10 sec.
nu Show only one measured value on given timestamp
Example : http://{BeeClear}/**bc_current**?date=1435322813&duration=8
Returns:

{"time":1429039347,
"items":[{"d":1428995914,"u":0.164,"g":0.000},
{"d":1428995924,"u":0.169,"g":0.000},
.....
{"d":1428997694,"u":0.233,"g":0.000},
{"d":1428997704,"u":0.235,"g":0.000}]}

Example : http://{BeeClear} /bc_current?nu=1451511279
{"d":1451511291,"time":20,"ul":2325500,"uh":1731113,"gl":597441,"gh":1545517,"u":0.495,"g":0.000,"gas":1589724}

3. Gas usage

Explanation : This command returns **recorded** historic gas usage values.

Command : **bc_usageGas**
Options : **date**|**duration**|**period**
Options Explained :
date (unix epoch time), set the ending date/time of measurement.
duraton = interval presentation in units of 1 hour
period = <day|week|month|year|number> time offset between values, number is in hours, when undefined.
Example : http://{BeeClear}/bc_usageGas?duraton=168&date=1426838400&period=6
Returns:

{"items":[
{"d":1426838400,"s":"473030323335363132313734333935313 4","g":1036269},
{"d":1426860000,"s":"473030323335363132313734333935313 4","g":1038399},
...
{"d":1427421600,"s":"473030323335363132313734333935313 4","g":1073991},
{"d":1427443200,"s":"473030323335363132313734333935313 4","g":1075460}]}

4. Smartmeter ID

Explanation : This commands returns the serial number of the gas and electricity meter. It also returns start and end date of registering.

Command : **bc_serials?**
Options : ---
Example : http://{BeeClear}/bc_serials
Returns :

{"serials": [
{"type": "g", "serial": "473030323335363132313734333935313 4", "start": 1424014245, "end": 0},
{"type": "e", "serial": "453030303931303030303830363937313 4", "start": 1424014245, "end": 0}]}

5. Read current installed update on device

Explanation : Check current installed software version on BeeClear
www = version web engine
firmware = version firmware
engine = version data collector / web server

Command : **bc_softwareVersion**
Options : ---
Example : http://{BeeClear}/bc_softwareVersion
Returns :

{"www": "","firmware": "","engine": "0.1.45"}

6A. Check, Download & Update software on device.

Explanation : a. Check for available software updates on update server. b. Download updates from update server. and c. Update local device with downloaded software

Command : **bc_update**
Options : **status | check | download | update**
Example : http://{BeeClear}/bc_update?cmd=status
Returns : with option **check** followed by **status**:

{"from":"check","info":"start"}
{"from":"check","info":"ok","firmware":"1.1.2"}
{"from":"check","info":"nok"}

With option **download** followed by **status**:
{"from":"download","info":"start"}
{"from":"download","info":"ok"}
{"from":"download","info":"nok"}
{"from":"download","info":"corrupt"}

With option **update** followed by **status**:
{"from":"update","info":"ok"}
{"from":"update","info":"nok"}

6B. Check for update

Explanation : Check latest available BeeClear update from internet.
Option : **check** (followed by option **status**).
Example : http://{BeeClear}/bc_update?cmd=check
Returns:
{"info":"ok"} -> command executed
{"info":"nok"} -> command not executed

Explanation : Use option **status** to return information
Example : http://{BeeClear}/bc_update?cmd=status
Returns :
{"from":"check","info":"start"}
{"from":"check","info":"ok","firmware":"1.1.2"}
{"from":"check","info":"nok"}

6C. Download update

Explanation : Download latest available BeeClear update version from internet.
Option : **download** (followed by option **status**).
Example : http://{BeeClear}/bc_update?cmd=download
Returns :
{"info":"ok"} -> command executed
{"info":"nok"} -> command not executed

Explanation : Use option **status** to return information
Example : http://{BeeClear}/bc_update?cmd=status
Returns :
{"from":"update","info":"ok"}
{"from":"update","info":"nok"}

6D. Update device & keep settings

Explanation : Update device with downloaded software version.
Option : **update** (followed by option **status**).
Example : http://{BeeClear}/bc_update?cmd=update
Returns :
{"info":"ok"} -> command executed
{"info":"nok"} -> command not executed

Explanation : Use option **status** to return information
Example : http://{BeeClear}/bc_update?cmd=status
Returns :
{"from":"update","info":"ok"}
{"from":"update","info":"nok"}

6E. Update device & erase settings

Explanation : Update device with downloaded software version and go to factory default.
Option : **update** option **erase**(followed by option **status**).
Example : http://{BeeClear}/bc_update?cmd=update&opts=erase
Returns :
{"info":"ok"} -> command executed
{"info":"nok"} -> command not executed

Explanation : Use option **status** to return information
Example : http://{BeeClear}/bc_update?cmd=status
Returns :
{"from":"update","info":"ok"}
{"from":"update","info":"nok"}

Location and content of files:
info update file:
http://beeclear.nl/updates/update.info
FIRMWARE=1.45
MD5SUM=019caf2d80d5523c9cdbb99f59c416f7
WEB_MD5SUM=019caf2d80d5523c9cdbb99f59c416f7

Binary firmware file:
http://beeclear.nl/updates/firmware.${FIRMWARE}.bin
Website file:
http://beeclear.nl/updates/www.${FIRMWARE}.tar.gz
Needs to be a compressed tar in a directory. Filename should look like:
www.123.tar.gz
www.123/index.html
www.123/favicon.ico

7A. Setup / Tariffs / getEnergieSettings

Explanation : This command shows the content of data energy containers. The "tarief" container is a list of detailed electricity costs. The "wie" container is a list of energy companies. Per list/entry in the "wie" container, there are two items/elements that reference the "tarief" container for energy costs. The two elements are **gas** and **electricity**. "**wie**":[{"naam":"nuon","**elec":0,"gas":3**,…}]. Of the "wie" container element 0 is the company name. Element 1 references **electricity** and element 2 references **gas**. The referenced tables for gas and electricity are separate mostly. The "wie" container is filled by the user via the setup interface (tarieven) in the GUI. Prerequisite: There needs to be at least one dataset/table in both containers: "tarief" and "who". You need to first run **put**EnergieSettings to fill at least one dataset in dataset table:"tarief" and one dataset in dataset table:"wie".
In short: Returns two tables: tarief and wie (tarief= lists of detailed energy costs. Wie= List of Energycompanies with gas and electricity references to tarief).

Command : **bc_getEnergieSettings**
Options : **---**
Example : http://{BeeClear}/bc_getEnergieSettings
Returns :

{"**tarief**":[{"tarief":0.010000,"DalTarief":0.020000,"AansluitingNetbeheerder":0.010000,"PeriodiekNetbeheerder":0.020000,"VastrechtAansluiting":0.010000,"VastrechtTransport":0.020000,"Systeemdiensten":0.010000,"Meetdiensten":0.020000,"energieBelasting":0.010000,"duurzameOpslag":0.020000},
{"tarief":0.010000,"DalTarief":0.020000,"AansluitingNetbeheerder":0.010000,"PeriodiekNetbeheerder":0.020000,"VastrechtAansluiting":0.010000,"VastrechtTransport":0.020000,"Systeemdiensten":0.010000,"Meetdiensten":0.020000,"energieBelasting":0.010000,"duurzameOpslag":0.020000},

{"tarief":0.010000,"DalTarief":0.020000,"AansluitingNetbeheerder":0.010000,"PeriodiekNetbeheerder":0.020000,"VastrechtAansluiting":0.010000,"VastrechtTransport":0.020000,"Systeemdiensten":0.010000,"Meetdiensten":0.020000,"energieBelasting":0.010000,"duurzameOpslag":0.020000},
{"tarief":0.010000,"DalTarief":0.020000,"AansluitingNetbeheerder":0.010000,"PeriodiekNetbeheerder":0.020000,"VastrechtAansluiting":0.010000,"VastrechtTransport":0.020000,"Systeemdiensten":0.010000,"Meetdiensten":0.020000,"energieBelasting":0.010000,"duurzameOpslag":0.020000},
{"tarief":0.900000,"DalTarief":0.800000,"AansluitingNetbeheerder":0.700000,"PeriodiekNetbeheerder":0.600000,"VastrechtAansluiting":0.500000,"VastrechtTransport":0.400000,"Systeemdiensten":0.300000,"Meetdiensten":0.200000,"energieBelasting":0.100000,"duurzameOpslag":0.123000},
{"tarief":0.900000,"DalTarief":0.800000,"AansluitingNetbeheerder":0.700000,"PeriodiekNetbeheerder":0.600000,"VastrechtAansluiting":0.500000,"VastrechtTransport":0.400000,"Systeemdiensten":0.300000,"Meetdiensten":0.200000,"energieBelasting":0.100000,"duurzameOpslag":0.123000}],
"**wie**":[{"naam":"nuon","**elec":0,"gas":3**,"btw":0.010000,"terug":0.020000,"toeslag":0.010000,"diverse":0.020000},{"naam":"eneco","elec":2,"gas":1,"btw":0.010000,"terug":0.020000,"toeslag":0.010000,"diverse":0.020000},
{"naam":"jojo","elec":4,"gas":5,"btw":0.300000,"terug":0.400000,"toeslag":0.500000,"diverse":0.600000}]}

Explanation:
There are two tables: “tarief” and “wie” (“energy rates” and “Who”). The tables “tarief” and “who” are filled with setEnergieSettings. Caution: Per energy unit Gas or Electricity two different tables should be referenced.

Example company nuon:
“nuon” from table “wie” (first entry in table):
naam":"nuon","**elec":0,"gas":3**,"btw":0.010000,"terug":0.020000,"toeslag":0.010000,"diverse":0.020000

*The “tarief” data for **electricity** will consist of (referenced by 2nd element “elec”:0):*
{"tarief":0.010000,"DalTarief":0.020000,"AansluitingNetbeheerder":0.010000,"PeriodiekNetbeheerder":0.020000,"VastrechtAansluiting":0.010000,"VastrechtTransport":0.020000,"Systeemdiensten":0.010000,"Meetdiensten":0.020000,"energieBelasting":0.010000,"duurzameOpslag":0.020000}

*The “tarief” data for **gas** will consist of (referenced by 3rd element “gas”:3):*
{"tarief":0.010000,"DalTarief":0.020000,"AansluitingNetbeheerder":0.010000,"PeriodiekNetbeheerder":0.020000,"VastrechtAansluiting":0.010000,"VastrechtTransport":0.020000,"Systeemdiensten":0.010000,"Meetdiensten":0.020000,"energieBelasting":0.010000,"duurzameOpslag":0.020000}

Value (Electricity)	GUI (SETUP – Tarieven)
Verbruik piektarief 0,00000 €/KWh	Tarief = “Verbruik piektarief” (**tarief**)
Verbruik daltarief 0,00000 €/KWh	DalTarief = “Verbruik daltarief” (**DalTarief**)
Capaciteitstarief aansluiting 0,00000 €/dag	AansluitingNetbeheerder= “Capaciteitstarief aansluiting”
Capaciteitstarief netbeheerder 0,00000 €/dag	PeriodiekNetbeheerder= “Capaciteitstarief netbeheerder”
Vastrecht aansluiting 0,00000 €/dag	VastrechtAansluiting = “Vastrecht aansluiting”
Vastrecht transport 0,00000 €/dag	VastrechtTransport = “Vastrecht transport”
Systeemdiensten 0,00000 €/dag	Systeemdiensten= “Systeemdiensten”
Meetdiensten 0,00000 €/dag	Meetdiensten= “Meetdiensten”
Duurzame opslag Electriciteit 0,00000 €/KWh	duurzameOpslag = “Duurzame opslag Elektriciteit”
Energiebelasting electriciteit 0,00000 €/KWh	energieBelasting = “Energiebelasting Elektriciteit”
BTW 21 %	btw

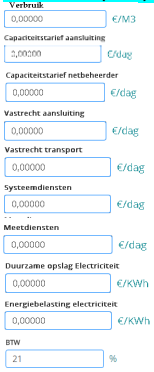
Contractdays = (contractday now) – (contractday start)

BtwElectricityHigh = (**Tarief**+ AansluitingNetbeheerder*contractdays + PeriodiekNetbeheerder*contractdays + VastrechtAansluiting*contractdays + VastrechtTransport*contractdays + Systeemdiensten*contractdays + Meetdiensten*contractdays + energieBelasting + duurzameOpslag
+ diverse) /100* btw

ElectrPriceUnitUsageHigh(uh)= (**Tarief**+ AansluitingNetbeheerder + PeriodiekNetbeheerder + VastrechtAansluiting + VastrechtTransport + Systeemdiensten + Meetdiensten + energieBelasting + duurzameOpslag+ diverse) + **BtwElectricityHigh**

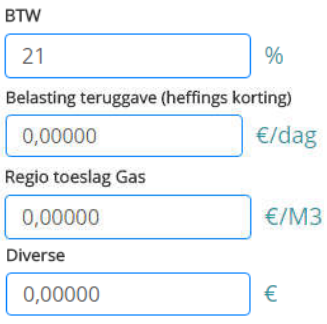
To calculate Electricity price in presentation UI's:
BtwElectricityLow = (DalTarief+ AansluitingNetbeheerder + PeriodiekNetbeheerder + VastrechtAansluiting + VastrechtTransport + Systeemdiensten + Meetdiensten + energieBelasting + duurzameOpslag + diverse)/100*Btw

ElectrPriceUnitUsageLow(ul)= (**DalTarief+** AansluitingNetbeheerder + PeriodiekNetbeheerder + VastrechtAansluiting + VastrechtTransport + Systeemdiensten + Meetdiensten + energieBelasting + duurzameOpslag + diverse) + **BtwElectricityLow**

Value (Gas)	GUI (SETUP – Tarieven)
	Tarief = "Verbruik Gas" (**normal** tariff)
	AansluitingNetbeheerder= "Capaciteitstarief aansluiting"
	PeriodiekNetbeheerder= "Capaciteitstarief netbeheerder"
	VastrechtAansluiting = "Vastrecht aansluiting"
	VastrechtTransport = "Vastrecht transport"
	Systeemdiensten= "Systeemdiensten"
	Meetdiensten= "Meetdiensten"
	duurzameOpslag = "Duurzame opslag Elektriciteit"
	energieBelasting = "Energiebelasting Elektriciteit"
	btw

GasBTW= (**DalTarief**+ AansluitingNetbeheerder + PeriodiekNetbeheerder + VastrechtAansluiting + VastrechtTransport + Systeemdiensten + Meetdiensten + energieBelasting + duurzameOpslag
+ toeslag + diverse)/100*btw

GasPriceUnit(gp)= (**DalTarief**+ AansluitingNetbeheerder + PeriodiekNetbeheerder + VastrechtAansluiting + VastrechtTransport + Systeemdiensten + Meetdiensten + energieBelasting + duurzameOpslag
+ toeslag + diverse) + GasBTW

Value Global (all tables the same value)	GUI
btw	
terug	
toeslag	
diverse	

- name = name of energy company
- elec = Electricity index refers to element number of dataset in tarief array.
- gas= Gas index refers to element number of dataset in tarief array.

- btw= tax in percentage
- terug= “Belasting teruggave / heffings korting” (tax return)
- toeslag= “regio toeslag gas” (region based costs)
- diverse= “Diverse” (variable extra costs)

7B. Setup / Tariffs / putEnergieSettings

Explanation : This command fills the information containers: “wie” and “tarief”. The “wie” container holds lists of the energy company information. The tarief conainter holds lists of the corresponding gas and electricity data.
The containers are filled by entering data for: “wie”,followed by the corresponding electricity and gas data.

Command : **bc_putEnergieSettings**
Options :
wie=[Name-Company_
btw_
energieTeruggave _
regioToeslag _
diverse]
elec=[tarief _
DalTarief _
AansluitingNetbeheerder _
PeriodiekNetbeheerder _
VastrechtAansluiting _
VastrechtTransport _
Systeemdiensten _
Meetdiensten _
energieBelasting _
duurzameOpslag]

Example : http://{BeeClear}/bc_putEnergieSettings&wie=Energiebedrijf 0.1 0.2 0.3 04&elec=0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0&gas=1.0 0.9 0.8 0.7 0.6 0.5 0.4 0.3 0.2 0.1

8A. Setup WiFi & Ethernet with bc_putNetwork

Explanation : This command will set network settings for the WIFI or Ethernet (Lan) device. With the **dev** option you choose which networking device needs configuring. The option **mode** will give you the choice of operating mode. Operating modes are “off” (turn device off), “static” (configure a static IP) and “dhcp” (configure automatic IP-address acquirement). With the extra options:
- Hostname you will be configuring the name of the host.
- IP you need to give the IPv4 bytes

Command : **bc_putNetwork**
Options :
dev =**ethernet | wifi**
proto =**off|static|dhcp**
Example : http://{BeeClear}/ bc_putNetwork&dev=wifi&mode=dhcp&hostname=beeclear &mode=ap&ssid=TEST&key=key1234
Returns :

Options: dev =**Ethernet | wifi**
proto =**off | static | dhcp**
hostname= $string hostname
ip= $ip version 4
netmask= $netmask
router= $route / default gw
dns= $dns

mode=**ap**|**client**

ssid= $Secure SID
passwd= $Wifi key

option dev=Ethernet.

mode=off,static,dhcp is only for ip interface
mode=off,ap,client is only for wifi antenne.

returns: {status:"ok"}" or {status:"fail"}
Examples:
http://beeclear/get-data?type=putNetwork&dev=ethernet&mode=off&hostname=beeclear
http://beeclear/get-data?type=putNetwork&dev=ethernet&mode=dhcp&hostname=beeclear
http://beeclear/get-data?type=putNetwork&dev=ethernet&mode=static&hostname=beeclear&ip=192.168.11.39&netmask=255.255.255.0

http://beeclear/get-data?type=putNetwork&dev=wifi&mode=off&hostname=beelear&mode=off&ssid=Wifi Cafe&key=passwd
http://beeclear/get-data?type=putNetwork&dev=wifi&mode=dhcp&hostname=beelear&mode=client&ssid=Wifi
Cafe&key=passwd
http://beeclear/get-data?type=putNetwork&dev=wifi&mode=static&ip=10.0.0.1&netmask=255.255.0.0&hostname=beelear&mode=ap&ssid=jojo44&key=12345678

Requirement to make it work in the GUI, you need two files:

/etc/config/network.eth
/etc/config/network.wifi with the correct settings.

8B. Read WIFI & Ethernet with bc_getNetwork

Explanation : This command retrieves the relevant networking settings for Ethernet and wifi. Settings like dns, default rout, Wifi encryption mode etc.
Command : **bc_getNetwork**
Example : http://{BeeClear} /bc_getNetwork
Returns :

{"status":"start"}

Run second time (poll):

{"status":"ok","scan":[{"dev":"wifi","ip":"192.168.111.1","netmask":"255.255.255.0","router":"","dns":"192.168.11.1"}
,{"dev":"ethernet","ip":"192.168.11.218","netmask":"255.255.255.0","router":"192.168.11.1","dns":"192.168.11.1"}
,{"dev":"config_wifi","ip":"192.168.111.1","netmask":"255.255.255.0","proto":"static","hostname":"beeclear","mode":"ap"
,"ssid":"BeeClear","key":"xmeter00","router":"","dns":""}
,{"dev":"config_ethernet","proto":"dhcp","hostname":"beeclear","router":"","dns":""}

9. Setup / WIFI / wifiScan

Explanation : Scans the WIFI-network for WIFI access points. The function returns a list of WIFI AP's and returns a list with corresponding signal strengths. The higher the dBm number indicates a stronger SSID signal. If the SSID is empty it is a "hidden" access point.
Command : bc_wifiScan
Example : http://{BeeClear}/bc_wifiScan
Returns :

{"status":"ok","scan":[{"ssid": "Cafe Duckstad","signal":" -76 dBm Quality","encryption":" WPA2 PSK (TKIP, CCMP)"}
,{"ssid": "KaRiN-DrAAdloos","signal":" -80 dBm Quality","encryption":"
mixed WPA/WPA2 PSK (TKIP, CCMP)"}
,{"ssid": "Dennis Prive Wifi","signal":" -82 dBm Quality","encryption":"

WPA2 PSK (TKIP)"}
,{"ssid": "arsj","signal":" -82 dBm Quality","encryption":" mixed
WPA/WPA2 PSK (TKIP, CCMP)"}
,{"ssid": "Honselersdijk","signal":" -88 dBm Quality","encryption":" WPA PSK (TKIP)"}
,{"ssid": "SEC_LinkShare_04ed87","signal":" -92 dBm Quality","encryption":" WPA2 PSK (CCMP)"}
,{"ssid": "Kraaijenest","signal":" -92 dBm Quality","encryption":" WPA2 PSK (CCMP)"}
,{"ssid": "F.B.I.","signal":" -94 dBm Quality","encryption":" WPA PSK (TKIP)"}
,{"ssid": "link","signal":" -88 dBm Quality","encryption":" WEP Open/Shared (NONE)"}]}

~~10. Set MeterType detection~~

~~Explanation : Sets behaviour of autodetection of metertype.~~

~~Command : bc_metertype~~
~~Example : http://{BeeClear}/bc_metertype?type=1~~
~~Properties :~~
~~metertype=-1 // Detect only once~~
~~metertype=-2 // autodetect always~~
~~metertype=3 // Type=3~~
~~metertype=4 // Type=4~~
~~Default=-1~~

~~Returns :~~
~~*{"info":"ok"} of {"info":"nok"}*~~

Setup & Misc

11. Poweroff

Explanation : Shutdown BeeClear and flush cash to disk.
Command : http://{BeeClear}/bc_poweroff (two times in one second)
Returns :

12. SaveConfig

Explanation : write configuration
Command : http://{BeeClear}/bc_saveconfig
Returns :

13A. Security

Explanation : Clears the session ID.
Command : http://{BeeClear}/bc_logout
Returns :
{"info":"ok"}

13B. Enable/disable Security

Explanation: Sets / Disables session security.
Command : http://{BeeClear}/bc_security?set=off
http://{BeeClear}/bc_security?set=on
Returns :
{"info":"nok"} or
{"info":"ok"}

14. MeterType

Explanation : Sets the DSMR meter type. Type three (3) or four (4). Sets metertype 4 and NTP to off. Sets metertype 3 with ntpd.
Command : http://{beeclear}/bc_meter3?set=off
http://{beeclear}/bc_meter3?set=on
Returns :
{"info":"nok"} or
{"info":"ok"}

15. Loglevel

Explanation : Sets the loglevel for httpd and serial requests.
Command : http://{beeclear}/bc_log?loglevel=0&logtype=512
Returns :
{"info":"nok"} or
{"info":"ok"}

16. RawLog

Explanation : Sets the rawlogging telegrams
Command : http://beeclear/bc_setting?type=rawlogging&set=off
http://beeclear/bc_setting?type=rawlogging&set=on
Returns :
{"info":"nok"} or
{"info":"ok"}

17. Debug

18. Log

19. Reset

20. Password

21. ServerTime