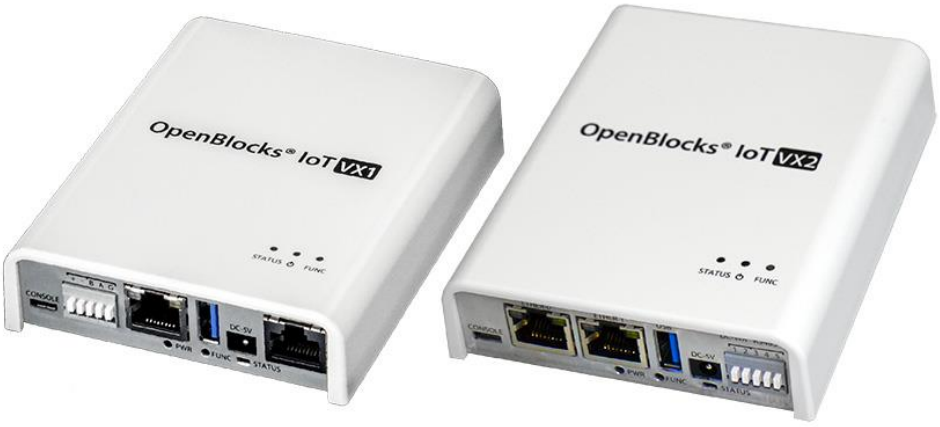
Plat' Home
TECHNOLOGY to serve you.

OpenBlocks IoT Family
Data Handling Configuration
Reference Guide



Ver.3.1.0

Plat'Home Co., Ltd.

■ About trademarks

- Company names, product names and other names in this document may be trademarks or registered trademarks of each company.
- Other proper names such as product names in this document are trademarks or registered trademarks of each company.

■ Before use

- It is prohibited to reprint the contents of this document in whole or in part without prior consent.
- The contents of this document are subject to change without notice.
- Although we make our best efforts to ensure the accuracy of this document, please contact our support center if you find typographical errors or other mistakes.
  The latest version of this document can be downloaded from our website.
- Note in advance that this device is not assumed to be used in in fields with fatal dangers.
- Note in advance that we bear no responsibility for any damages and lost earnings resulting from operation of this device.

Table of Contents

1. Common Subject Matter

1.1. Module

Applications such as PD Repeater, PD Handler, PD Agent, and PD Broker that perform data handling are called **module**.

2.1. Configuration File

1.2.1. Format

The configuration file of each module is described in JSON format.

1.2.2. Default Value

Each module has a default value for parameters (in the following description, it is called a **key**.) that can be set in the configuration file unless otherwise specified.

3.1. Inter-Module Communication

1.3.1. Method

Unix domain socket is used for data transfer between modules.

1.3.2. Path Name of Socket

The path name of the socket that receives the data is the **bind** key, and the path name of the destination socket of the data can be changed with the **push_to** key.
If the socket's path name begins with '@', it is interpreted as Abstract name space.

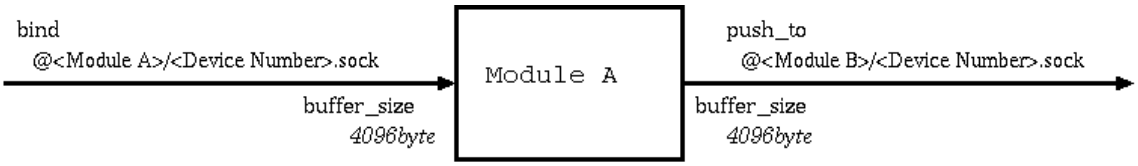
1.3.3. Data Size

The default buffer size of data is 4,096 bytes. In the module that receives data, you can change it with the **buffer_size** key.

1.3.4. Key for configuer of communication between modules.

Key	Data Type	Description
bind	Strings	The name of the socket that receives data, interpreted as abstract namespace if the beginning of the string is '@', default values for each module are used if empty except pd_broker. It is specified in "@ / <*module name*> / <*device number*> .sock" format.
push_to	Strings	The destination socket name of the data, interpreted as abstract namespace if the beginning of the string is '@', default values for each module are used if empty except pd_broker. It is specified in "@ / <*module name*> / <*device number*> .sock" format.
buffer_size	Integer	Data buffer size (byte)

Key for configuer of communication between modules.



2. PD Repeater

1.1. Default Path Name

The default path name of files related to PD Repeater is as follows.

Path Name	Description
/usr/sbin/pd_repeater	Resident execution object (server)
/usr/sbin/pd_repeater_p	Execution object (utility)
/lib/systemd/system/pd_repeater.service	Systemd servicefile
/etc/init.d/pd_repeater	RC file
/var/webui/config/pd_repeater.conf	Configration file
/var/run/pd_repeater.pid	PIDfile
/var/webui/pd-data/pd_repeater.db	Cache file
@/pd_repeater/<*device number*>.sock	UNIX domain socket for input.
@/pd_handler/<*device number*>.sock	UNIX domain socket for output.

Default path name of files related to PD Repeater.

2.1. Format of Configuration File

2.2.1. Syntax

```
{
    "db_file" : "<cache file>",
    "max_db_size" : <cache size>,
    "servers": {
        "<cloud index key>" : {
            <JSON object for server configuration>
        },
        "< cloud index key >" : {
            <JSON object for server configuration>
        }
      },
      "devices": [
          {
              "localname" : "<device number>",
              "bind" : "<UNIX domain socket for input>",
              "push_to" : "<UNIX domain socket for output>",
              "buffer_size" : <data buffer size >,
              "truncate" : <time span to thin out data>,
              "receive" : <boolean value>,
              "< cloud index key >" : {
                  <JSON object for cloud specific device configuration>
              },
              "<cloud index key>" : {
                  <JSON object for cloud specific device configuration>
              }
          }
      ]
  }
```

2.2.2. Root Object

Key	Data Type	Description
db_file	Strings	Path name of the cache file. Default is '/var/webui/pd-data/pd_repeater.db'. (MAXPATHLEN)
max_db_size	Integer	Maximum cache file size (Mbyte). Default is 16.
servers	JSON obj	servers Object
devices	JSON array	devices Object

Root Object

2.2.3. servers Object

Key	Data Type	Description
pd_ex	JSON obj	Configuration object for sending data to PD Exchange.
mqtt	JSON obj	Configuration object for sending data to Generic MQTT broker.
w4d	JSON obj	Configuration object for sending data to IBM Watson IoT for device.
kinesis	JSON obj	Configuration object for sending data to Amazon Kinesis.
pgsql	JSON obj	Configuration object for sending data to PostgreSQL database.
mysql	JSON obj	Configuration object for sending data to MySQL database.
web	JSON obj	Configuration object for sending data to Generic Web server.
awsiot	JSON obj	Configuration object for sending data to Amazon AWS IoT.
eventhub	JSON obj	Configuration object for sending data to Microsoft Azure Event Hubs.
w4g	JSON obj	Configuration object for sending data to IBM Watson IoT for gateway.
iothub	JSON obj	Configuration object for sending data to Microsoft Azure IoT Hub.
t4d	JSON obj	Configuration object for sending data to NTT docomo Toami for docomo.
lsocket	JSON obj	Configuration object for sending data to Unix domain socket.
kddi_std	JSON obj	Configuration object for sending data to KDDI IoT cloud Standard.
nf_dvhub	JSON obj	Configuration object for sending data to NIFTY IoT Device Hub.
pd_web	JSON obj	Configuration object for sending data to PD Web(Plat 'Home proprietary Web Server).
iotcore	JSON obj	Configuration object for sending data to Google Cloud IoT Core.
ltcp	JSON obj	Configuration object for sending data to TCP socket.

servers Object

By using a character string in which the same character string is appended with '_ 0' or '_ 1' as a cloud index key('pd_ex_ 0', 'pd_ex_ 1' for 'pd_ex', for example), two different servers (or endpoint) in the same cloud can be set.

2.2.3.1. PD Excheng(pd_ex)

Key	Data Type	Description
enable	Boolean	Default is false
interval	Integer	Interval time to send data(sec). Default is 30.
expier	Integer	Retention time of data(sec). Default is 0, it is unlimited until data is transmitted.
restart	Integer	Interval time to restart sub-process(sec). Default is 604800, if it is 0, do not restart.
url	Strings	Destination URL. Default is 'http://pd.plathome.com'. (MAXPATHLEN)
poll_interval	Integer	Interval time of polling command that message in the downstream direction from PD Exchange(sec). Default is 30.
deid_prefix	Strings	PD Exchange device ID prefix. (10byte)
secretkey	Strings	Secretkey for connection. (16byte)

Configuration Object for PD Exchange (pd ex)

2.2.3.2. Generic MQTT Broker(mqtt)

Key	Data Type	Description
enable	Boolean	Default is false
interval	Integer	Interval time to send data(sec). Default is 30.
expier	Integer	Retention time of data(sec). Default is 0, it is unlimited until data is transmitted.
restart	Integer	Interval time to restart sub-process(sec). Default is 604800, if it is 0, do not restart.
host	Strings	Specify the destination host by IP address or FQDN. (MAXHOSTNAMELEN)
protocol	Strings	Connection protocol,specify 'tcp' or 'ssl' . Default is 'tcp'.
port	Integer	Connection port number. Default is 1883 for ‘tcp,8883 for ‘ssl’.
keepaliveinterval	Integer	Keepalive interval of MQTT (sec). Default is 10.
cleansession	Boolean	Clean session of MQTT. Default is true.
reliable	Boolean	Reliable of MQTT. Default is true.
qos	Integer	Transmission QoS of MQTT, Specify 0,1or2. Default is 0.
retained	Boolean	Retained of MQTT. Default is false.
rcv_qos	Integer	Receive QoS of MQTT, Specify 0,1or2. Default is 1.
client_id	Strings	Client ID for MQTT connection. (64byte)
topic_prefix	Strings	A character string appended to the top of the MQTT transmission topic. The sending topic is a combination of this character string and the character string specified in the device setting object 'unique_id', which is *topic_ prefix*/*unique_id*. (MAXPATHLEN)
rcv_topic_prefix	Strings	A character string appended to the beginning of the receive topic of MQTT. Receive topics, it stands by in *rcv_topic_prefix*/# and transfers the received message internally to a device that completely matches *rcv_topic_prefix*/*unique_id* consisting of a combination of character strings specified in the device setting object 'unique id'. If they do not exactly match, the received message is transferred to a plurality of devices that match by forward match. (MAXPATHLEN)
username	Strings	Username for MQTT connection. (32byte)
password	Strings	Password for MQTT password authentication. (128byte)
truststore	Strings	Path name of Root CA certificate file for SSL connection. (MAXPATHLEN)
keystore	Strings	Path name of certificate file for SSL connection. (MAXPATHLEN)
privatekey	Strings	Path name of privatekey file for SSL connection. (MAXPATHLEN)

Configuration Object for Generic MQTT Broker(mqtt)

2.2.3.3. IBM Watson IoT for device (w4d)

Key	Data Type	Description
enable	Boolean	Default is false
interval	Integer	Interval time to send data(sec). Default is 30.
expier	Integer	Retention time of data(sec). Default is 0, it is unlimited until data is transmitted.
restart	Integer	Interval time to restart sub-process(sec). Default is 604800, if it is 0, do not restart.
domain	Strings	Destination domain name. Default is 'messag-ing.internetofthings.ibmcloud.com'. (MAXHOSTNAMELEN)
protocol	Strings	Connection protocol,specify 'tcp' or 'ssl' . Default is 'tcp'.
port	Integer	Connection port number. Default is 1883 for ‘tcp,8883 for ‘ssl’.
keepaliveinterval	Integer	Keepalive interval of MQTT (sec). Default is 10.
cleansession	Boolean	Clean session of MQTT. Default is true.
reliable	Boolean	Reliable of MQTT. Default is true.
qos	Integer	Transmission QoS of MQTT, Specify 0,1or2. Default is 0.
retained	Boolean	Retained of MQTT. Default is false.
rcv_qos	Integer	Receive QoS of MQTT, Specify 0,1or2. Default is 1.
org_id	Strings	Organization ID. Default is 'quickstart'. (16byte)
event_id	Strings	Event ID. Default is 'sample'. (128byte)
format_string	Strings	Data format. Default is 'json'. (16byte)
truststore	Strings	Path name of Root CA certificate file for SSL connection. (MAXPATHLEN)

Configuration Object for IBM Watson IoT for device (w4d)

2.2.3.4. Amazon Kinesis (kinesis)

Key	Data Type	Description
enable	Boolean	Default is false
interval	Integer	Interval time to send data(sec). Default is 30.
expier	Integer	Retention time of data(sec). Default is 0, it is unlimited until data is transmitted.
restart	Integer	Interval time to restart sub-process(sec). Default is 604800, if it is 0, do not restart.
domain	Strings	Destination domain name. Default is 'amazonaws.com'. (MAXHOSTNAMELEN)
region	Strings	Destination region name. Default is 'ap-northeast-1'. (64byte)
accessid	Strings	Access ID. (128byte)
accesskey	Strings	Access key. (128byte)
streamname	Strings	Stream name. (128byte)

Configuration Object for Amazon Kinesis (kinesis)

2.2.3.5. PostgreSQL (pgsql)

Key	Data Type	Description
enable	Boolean	Default is false
interval	Integer	Interval time to send data(sec). Default is 30.
expier	Integer	Retention time of data(sec). Default is 0, it is unlimited until data is transmitted.
restart	Integer	Interval time to restart sub-process(sec). Default is 604800, if it is 0, do not restart.
hostaddr	Strings	Destination IP address. (MAXHOSTNAMELEN)
protocol	Strings	Connection protocol,specify 'tcp' or 'ssl' . Default is 'tcp'.
port	Integer	Connection port number. Default is 5432.
sslmode	Strings	SSL mode of PostgreSQL . Default is 'verify-full'. (12byte)
pg_type	Integer	OID defined in include file that postgresql / server / catalog / pg_type.h. Defaults is 25 (TEXTOID).
dbname	Strings	Database name. Default is 'pd repeater'. (32byte)
table	Strings	Name of table to write data to. Default is 'buf'. (32byte)
username	Strings	Username for connection. (32byte)
password	Strings	Password for connection. (32byte)
truststore	Strings	Path name of Root CA certificate file for SSL connection. (MAXPATHLEN)
keystore	Strings	Path name of certificate file for SSL connection. (MAXPATHLEN)
privatekey	Strings	Path name of privatekey file for SSL connection. (MAXPATHLEN)

Configuration Object for PostgreSQL (pgsql)

2.2.3.6. MySQL (mysql)

Key	Data Type	Description
enable	Boolean	Default is false
interval	Integer	Interval time to send data(sec). Default is 30.
expier	Integer	Retention time of data(sec). Default is 0, it is unlimited until data is transmitted.
restart	Integer	Interval time to restart sub-process(sec). Default is 604800, if it is 0, do not restart.
hostaddr	Strings	Destination IP address. (MAXHOSTNAMELEN)
protocol	Strings	Connection protocol,specify 'tcp' or 'ssl' . Default is 'tcp'.
port	Integer	Connection port number. Default is 5432.
dbname	Strings	Database name. Default is 'pd repeater'. (32byte)
table	Strings	Name of table to write data to. Default is 'buf'. (32byte)
username	Strings	Username for connection. (32byte)
password	Strings	Password for connection. (32byte)
truststore	Strings	Path name of Root CA certificate file for SSL connection. (MAXPATHLEN)
keystore	Strings	Path name of certificate file for SSL connection. (MAXPATHLEN)
privatekey	Strings	Path name of privatekey file for SSL connection. (MAXPATHLEN)

Configuration Object for MySQL (mysql)

2.2.3.7. Generic Web Server(web)

Key	Data Type	Description
enable	Boolean	Default is false
interval	Integer	Interval time to send data(sec). Default is 30.
expier	Integer	Retention time of data(sec). Default is 0, it is unlimited until data is transmitted.
restart	Integer	Interval time to restart sub-process(sec). Default is 604800, if it is 0, do not restart.
url	Strings	Destination URL. (MAXPATHLEN)
username	Strings	Username for BASIC authentication, if empty, no authentication. (32byte)
password	Strings	Password for BASIC authentication, if empty, no authentication. (32byte)
max_post_msize	Integer	Maximum post size .(Mbyte). Default is 1.
content_type	Strings	Specify 'Content-Type'. if 'text/plain' or 'application/json' is specified, URL safe encoding of the payload is not performed. Defalut is empty. (64byte)

Configuration Object for Generic Web Server

2.2.3.8. Amazon AWS IoT (awsiot)

Key	Data Type	Description
enable	Boolean	Default is false
interval	Integer	Interval time to send data(sec). Default is 30.
expier	Integer	Retention time of data(sec). Default is 0, it is unlimited until data is transmitted.
restart	Integer	Interval time to restart sub-process(sec). Default is 604800, if it is 0, do not restart.
host	Strings	Destination hostname. (MAXHOSTNAMELEN)
protocol	Strings	Connection protocol,specify 'tcp' or 'ssl' . Default is 'tcp'.
port	Integer	Connection port number. Default is 8883.
keepaliveinterval	Integer	Keepalive interval of MQTT (sec). Default is 10.
reliable	Boolean	Reliable of MQTT. Default is true.
qos	Integer	Transmission QoS of MQTT, Specify 0,1or2. Default is 0.
retained	Boolean	Retained of MQTT. Default is false.
rcv_qos	Integer	Receive QoS of MQTT, Specify 0,1or2. Default is 1.
rootCA	Strings	Path name of Root CA certificate file for SSL connection. (MAXPATHLEN)

Configuration Object for Amazon AWS IoT (awsiot)

2.2.3.9. Microsoft Azure Event Hubs (eventhub)

Key	Data Type	Description
enable	Boolean	Default is false
interval	Integer	Interval time to send data(sec). Default is 30.
expier	Integer	Retention time of data(sec). Default is 0, it is unlimited until data is transmitted.
restart	Integer	Interval time to restart sub-process(sec). Default is 604800, if it is 0, do not restart.
domain	Strings	Destination domain name. Default is 'servicebus.windows.net'. (MAXHOSTNAMELEN)
namespace	Strings	Destination name space. (64byte)
port	Integer	Connection port number. Default is 5671.

Configuration Object for Microsoft Azure Event Hubs (eventhub)

2.2.3.10. IBM Watson IoT for gateway (w4g)

Key	Data Type	Description
enable	Boolean	Default is false
interval	Integer	Interval time to send data(sec). Default is 30.
expier	Integer	Retention time of data(sec). Default is 0, it is unlimited until data is transmitted.
restart	Integer	Interval time to restart sub-process(sec). Default is 604800, if it is 0, do not restart.
domain	Strings	Destination domain name. Default is 'messag-ing.internetofthings.ibmcloud.com'. (MAXHOSTNAMELEN)
protocol	Strings	Connection protocol,specify 'tcp' or 'ssl' . Default is 'tcp'.
port	Integer	Connection port number. Default is 1883 for ‘tcp,8883 for ‘ssl’.
keepaliveinterval	Integer	Keepalive interval of MQTT (sec). Default is 10.
cleansession	Boolean	Clean session of MQTT. Default is true.
reliable	Boolean	Reliable of MQTT. Default is true.
qos	Integer	Transmission QoS of MQTT, Specify 0,1or2. Default is 0.
retained	Boolean	Retained of MQTT. Default is false.
rcv_qos	Integer	Receive QoS of MQTT, Specify 0,1or2. Default is 1.
org_id	Strings	Organization ID. Default is 'quickstart'. (16byte)
gateway type	Strings	Gateway type. Default is 'sample'. (20byte)
gateway id	Strings	Gateway ID. (20byte)
event_id	Strings	Event ID. Default is 'sample'. (128byte)
format_string	Strings	Data format. Default is 'json'. (16byte)
truststore	Strings	Path name of Root CA certificate file for SSL connection. (MAXPATHLEN)
keystore	Strings	Path name of certificate file for SSL connection. (MAXPATHLEN)
privatekey	Strings	Path name of privatekey file for SSL connection. (MAXPATHLEN)

Configuration Object for IBM Watson IoT for gateway (w4g)

2.2.3.11. Microsoft Azure IoT Hub(iothub)

Key	Data Type	Description
enable	Boolean	Default is false
interval	Integer	Interval time to send data(sec). Default is 30.
expier	Integer	Retention time of data(sec). Default is 0, it is unlimited until data is transmitted.
restart	Integer	Interval time to restart sub-process(sec). Default is 604800, if it is 0, do not restart.
domain	Strings	Destination domain name. Default is 'azure-devices.net'. (MAXHOSTNAMELEN)
hub_name	Strings	Destination hub name. (64byte)
gw_host	Strings	Gateway hostname for IoT Edge.(MAXHOSTNAMELEN)
tail_slash	Boolean	"/" Insertion flag to the end of MQTT topic, default value is true.
keepaliveinterval	Integer	Keepalive interval of MQTT (sec). Default is 10.
cleansession	Boolean	Clean session of MQTT. Default is true.
reliable	Boolean	Reliable of MQTT. Default is true.
qos	Integer	Transmission QoS of MQTT, Specify 0,1or2. Default is 0.
retained	Boolean	Retained of MQTT. Default is false.
rcv_qos	Integer	Receive QoS of MQTT, Specify 0,1or2. Default is 1.
rootCA	Strings	Path name of Root CA certificate file used for X.509 authentication. (MAXHOSTNAMELEN)

Configuration Object for Microsoft Azure IoT Hub(iothub)

2.2.3.12. NTT docomo Toami for docomo (t4d)

Key	Data Type	Description
enable	Boolean	Default is false
interval	Integer	Interval time to send data(sec). Default is 30.
expier	Integer	Retention time of data(sec). Default is 0, it is unlimited until data is transmitted.
restart	Integer	Interval time to restart sub-process(sec). Default is 604800, if it is 0, do not restart.
url	Strings	Destination URL. Default is 'https://xxxx.to4do.com:443'. (MAX-PATHLEN)

Configuration Object for NTT docomo Toami for docomo (t4d)

2.2.3.13. UNIX Domain Soket (lsocket)

Key	Data Type	Description
enable	Boolean	Default is false
interval	Integer	Interval time to send data(sec). Default is 30.
expier	Integer	Retention time of data(sec). Default is 0, it is unlimited until data is transmitted.
restart	Integer	Interval time to restart sub-process(sec). Default is 604800, if it is 0, do not restart.
root_path	Strings	The root path name of the destination Unix domain socket. If the beginning of the string is '@', interpreted as abstract namespace . Default is '/tmp'. (MAX-PATHLEN)

Configuration Object for UNIX Domain Soket (lsocket)

2.2.3.14. KDDI IoT cloud Standard (kddi std)

Key	Data Type	Description
enable	Boolean	Default is false
interval	Integer	Interval time to send data(sec). Default is 30.
expier	Integer	Retention time of data(sec). Default is 0, it is unlimited until data is transmitted.
restart	Integer	Interval time to restart sub-process(sec). Default is 604800, if it is 0, do not restart.
domain	Strings	Destination domain name. Default is 'datalink.m2m-cloud-std.kddi.ne.jp'. (MAXHOSTNAMELEN)
port	Integer	Connection port number. Default is 443.
termid	Strings	Terminal ID. (16byte)
username	Strings	Username for BASIC authentication, if empty, no authentication. (32byte)
password	Strings	Password for BASIC authentication, if empty, no authentication. (32byte)

Configuration Object for KDDI IoT cloud Standard (kddi std)

2.2.3.15. NIFTY IoT Device Hub (nf_dvhub)

Key	Data Type	Description
enable	Boolean	Default is false
interval	Integer	Interval time to send data(sec). Default is 30.
expier	Integer	Retention time of data(sec). Default is 0, it is unlimited until data is transmitted.
restart	Integer	Interval time to restart sub-process(sec). Default is 604800, if it is 0, do not restart.
host	Strings	Destination hostname. Default is 'iot-device.jp-east-1.mqtt.cloud.nifty.com' (MAXHOSTNAMELEN)
protocol	Strings	Connection protocol,specify 'tcp' or 'ssl' . Default is 'tcp'.
keepaliveinterval	Integer	Keepalive interval of MQTT (sec). Default is 10.
reliable	Boolean	Reliable of MQTT. Default is true.
qos	Integer	Transmission QoS of MQTT, Specify 0,1or2. Default is 0.
rootCA	Strings	Path name of Root CA certificate file for SSL connection. (MAXPATHLEN)

Configuration Object for NIFTY IoT Device Hub (nf_dvhub)

2.2.3.16. PD Web (pd_web)

Key	Data Type	Description
enable	Boolean	Default is false
interval	Integer	Interval time to send data(sec). Default is 30.
expier	Integer	Retention time of data(sec). Default is 0, it is unlimited until data is transmitted.
restart	Integer	Interval time to restart sub-process(sec). Default is 604800, if it is 0, do not restart.
url	Strings	Destination URL. (MAXPATHLEN)
poll_interval	Integer	In the PD Web, it acquires the message in the downstream direction at the same time as sending the message, but assuming the case that the message to be sent does not exist in the poll interval, it specifies the interval (sec) at which the empty connection will be generated. Default is 30.
username	Strings	Username for BASIC authentication, if empty, no authentication. (32byte)
password	Strings	Password for BASIC authentication, if empty, no authentication. (32byte)
max_post_msize	Integer	Maximum post size .(Mbyte). Default is 1.

Configuration Object for PD Web / Plat 'Home proprietary Web Server (pd_web)

2.2.3.17. Google Cloud IoT Core (iotcore)

Key	Data Type	Description
enable	Boolean	Default is false
interval	Integer	Interval time to send data(sec). Default is 30.
expier	Integer	Retention time of data(sec). Default is 0, it is unlimited until data is transmitted.
restart	Integer	Interval time to restart sub-process(sec). Default is 604800, if it is 0, do not restart.
host	Strings	Destination hostname.(MAXHOSTNAMELEN)
port	Integer	Connection port number. Default is 8883.
project_id	Strings	Project ID. (32bytes)
cloud_region	Strings	Cloud region. (16bytes)
keepaliveinterval	Integer	Keepalive interval of MQTT (sec). Default is 10.
cleansession	Boolean	Clean session of MQTT. Default is true.
reliable	Boolean	Reliable of MQTT. Default is true.
qos	Integer	Transmission QoS of MQTT, Specify 0,1or2. Default is 0.
retained	Boolean	Retained of MQTT. Default is false.
rcv_qos	Integer	Receive QoS of MQTT, Specify 0,1or2. Default is 1.
rootCA	Strings	Path name of Root CA certificate file for SSL connection. (MAXPATHLEN)

Configuration Object for Google Cloud IoT Core (iotcore)

2.2.3.18. TCP Soket (ltcp)

Key	Data Type	Description
enable	Boolean	Default is false
interval	Integer	Interval time to send data(sec). Default is 30.
expier	Integer	Retention time of data(sec). Default is 0, it is unlimited until data is transmitted.
restart	Integer	Interval time to restart sub-process(sec). Default is 604800, if it is 0, do not restart.
ip_addr	Strings	Destination IP address. Default is '127.0.0.1'. (16byte)
delimiter	Strings	Message separator code. Default is '0x00'. For example, '0x0d0a' when CRLF code is used as a separator. (7byte)

Configuration Object for TCP Soket (ltcp)

2.2.4. devices Object

Key	Data Type	Description
localname	Strings	Local name of device(Device number). (32byte)
bind	Strings	The name of the socket that receives data, interpreted as abstract namespace if the beginning of the string is '@'. It is specified in “@ / <*module name*> / <*device number*> .sock” format. If empty, default value, @/pd_repeater/<*device number*> .sock is specified.
push_to	Strings	The destination socket name of the data, interpreted as abstract namespace if the beginning of the string is '@'. It is specified in “@ / <*module name*> / <*device number*> .sock” format. If empty, default value, @/pd_handler/<*device number*> .sock is specified.
buffer_size	Integer	Data buffer size (byte). Default is 4096.
truncate	Real	Time span to thin out data (msec). Default is 0.0, not thin out.
receive	Boolean	Whether to receive messages from the cloud in the downstream direction. Default is false
pd_ex	JSON obj	PD Exchenge specific configuration object.
mqtt	JSON obj	Genric MQTT Broker specific configuration object.
w4d	JSON obj	IBM Watson IoT for device specific configuration object.
kinesis	JSON obj	Amazon Kinesis specific configuration object.
pgsql	JSON obj	PostgreSQL database specific configuration object.
mysql	JSON obj	MySQL database specific configuration object.
web	JSON obj	Generic Web server specific configuration object.
awsiot	JSON obj	Amazon AWS IoT specific configuration object.
eventhub	JSON obj	Microsoft Azure Event Hubs specific configuration object.
w4g	JSON obj	IBM Watson IoT for gateway specific configuration object.
iothub	JSON obj	Microsoft Azure IoT Hub specific configuration object.
t4d	JSON obj	NTT docomo Toami for docomo specific configuration object.
lsocket	JSON obj	Unix domain socket specific configuration object.
kddi_std	JSON obj	KDDI IoT cloud Standard specific configuration object.
nf_dvhub	JSON obj	NIFTY IoT Device Hub specific configuration object.
pd_web	JSON obj	PD Web(Plat 'Home proprietary Web Server) specific configuration object.
iotcore	JSON obj	Google Cloud IoT Core specific configuration object.
ltcp	JSON obj	TCP socket specific configuration object.

devices Object

By using a character string in which the same character string is appended with '_ 0' or '_ 1' as a cloud index key('pd_ex_ 0', 'pd_ex_ 1' for 'pd_ex', for example), two different servers (or endpoint) in the same cloud can be set, as same as Servers object.

2.2.4.1. PD Exchenge(pd_ex)

Key	Data Type	Description
enable	Boolean	Default is false
deid suffix	Strings	Device ID suffix. (9byte)

PD Exchenge(pd_ex) specific configuration object

2.2.4.2. Generic MQTT Broker(mqtt)

Key	Data Type	Description
enable	Boolean	Default is false
unique_id	Strings	Unique ID. (20byte)

Generic MQTT Broker(mqtt) specific configuration object

2.2.4.3. IBM Watson IoT for device (w4d)

Key	Data Type	Description
enable	Boolean	Default is false
device_id	Strings	Device ID. (20byte)
device_type	Strings	Device type. (20byte)
password	Strings	Password. (128byte)
keystore	Strings	Path name of certificate file for SSL connection. (MAXPATHLEN)
privatekey	Strings	Path name of privatekey file for SSL connection. (MAXPATHLEN)

IBM Watson IoT for device (w4d) configuration object

2.2.4.4. Amazon Kinesis (kinesis)

Key	Data Type	Description
enable	Boolean	Default is false

Amazon Kinesis (kinesis) specific configuration object

2.2.4.5. PostgreSQL (pgsql)

Key	Data Type	Description
enable	Boolean	Default is false

PostgreSQL (pgsql) specific configuration object

2.2.4.6. MySQL (mysql)

Key	Data Type	Description
enable	Boolean	Default is false

MySQL (mysql) specific configuration object

2.2.4.7. Generic Web Server(web)

Key	Data Type	Description
enable	Boolean	Default is false

Generic Web Server(web) specific configuration object

2.2.4.8. Amazon AWS IoT (awsiot)

Key	Data Type	Description
enable	Boolean	Default is false
client_id	Strings	Client ID. (20byte)
thing_name	Strings	thingName for Device Shadow. (128byte)
topic	Strings	MQTT transmission topic. (MAXPATHLEN)
rcv_topic	Strings	MQTT receive topic. (MAXPATHLEN)
cert	Strings	Path name of certificate file for SSL connection. (MAXPATHLEN)
privatekey	Strings	Path name of privatekey file for SSL connection. (MAXPATHLEN)

Amazon AWS IoT (awsi) specific configuration object

2.2.4.9. Microsoft Azure Event Hubs (eventhub)

Key	Data Type	Description
enable	Boolean	Default is false
hub_name	Strings	Event Hubs 名. (64byte)
sas_policy	Strings	SAS policy. (64byte)
sas_key	Strings	SAS key. (64byte)

Microsoft Azure Event Hubs (eventhub) specific configuration object

2.2.4.10. IBM Watson IoT for gateway (w4g)

Key	Data Type	Description
enable	Boolean	Default is false
device_id	Strings	Device ID. (20byte)
device_type	Strings	Device type. (20byte)

IBM Watson IoT for gateway (w4g) specific configuration object

2.2.4.11. Microsoft Azure IoT Hub(iothub)

Key	Data Type	Description
enable	Boolean	Default is false
device_id	Strings	Device ID. (128byte)
device_key	Strings	Device key. (64byte)
module_id	Strings	Module ID for IoT Edge. (64byte)
cert	Strings	Path name of certificate file used for X.509 authentication. (MAXPATHLEN)
privatekey	Strings	Path name of privatekey file used for X.509 authentication. (MAXPATHLEN)

Microsoft Azure IoT Hub(iothub) specific configuration object

2.2.4.12. NTT docomo Toami for docomo (t4d)

Key	Data Type	Description
enable	Boolean	Default is false
gwname	Strings	GatewayName. (32byte)
appkey	Strings	AppKey. (64byte)

NTT docomo Toami for docomo (t4d) specific configuration object

2.2.4.13. UNIX Domain Soket (lsocket)

Key	Data Type	Description
enable	Boolean	Default is false

UNIX Domain Soket (lsocket) specific configuration object

2.2.4.14. KDDI IoT cloud Standard (kddi std)

Key	Data Type	Description
enable	Boolean	Default is false

KDDI IoT cloud Standard (kddi std) specific configuration object

2.2.4.15. NIFTY IoT Device Hub (nf_dvhub)

Key	Data Type	Description
enable	Boolean	Default is false
event_type	Strings	Event type. (20byte)
device_id	Strings	Device ID. (20byte)
device_api_key	Strings	Device API key(169byte)

NIFTY IoT Device Hub (nf_dvhub) specific configuration object

2.2.4.16. PD Web (pd web)

Key	Data Type	Description
enable	Boolean	Default is false
id	Strings	Client ID. (128byte)
key	Strings	String used for token authentication (access key). (128byte)

PD Web / Plat 'Home proprietary Web Server (pd_web) specific configuration object

2.2.4.17. Google Cloud IoT Core (iotcore)

Key	Data Type	Description
enable	Boolean	Default is false
registry_id	Strings	Registry ID. (32byte)
device_id	Strings	Device ID. (32byte)
jwt_algorithm	Strings	JWT algorithm. 'RS256' or 'ES256' .(6byte)
cert	Strings	Path name of certificate file for SSL connection. (MAXPATHLEN)
privatekey	Strings	Path name of privatekey file for SSL connection. (MAXPATHLEN)

Google Cloud IoT Core (iotcore) specific configuration object

2.2.4.18. TCP Soket (ltcp)

Key	Data Type	Description
enable	Boolean	Default is false
port	Strings	Connection port number. Default is 49152.

TCP Soket (ltcp) specific configuration object

2.3.Downstream Message

The format of the message that the PD Repeater forwards to the downstream module.

Cloud ID (1Byte)	Sub ID (1Byte)	Header Size (2Bytes)	MD5 (16Bytes)	Header	Payload

Field	Bytes	Description
Cloud ID	1	The number of the cloud to be sent and received assigned for convenience.
Sub ID	1	Identifier when using two servers in the same cloud. (0x00 or 0x01)
Header Size	2	Size of Header added by PD Reperater.
MD5	16	Hash value of downstream message sent from cloud (MD5)
Header	Variable	MQTT topic or HTTP response header decoded by PD Reperater.
Payload	Variable	downstream message sent from cloud.

Format of the Downstream Message

Cloud ID and the contents of the header.

Cloud	Cloud ID	Support Downstream Message	Contents of Header
PD Exchange	0x00	○	Command ID, ApplicationID
Generic MQTT Broker	0x01	○	MQTT receive topic
Watson IoT for Device	0x02	○	MQTT receive topic
Amazon Kinesis	0x03		
PostgreSQL database	0x04		
MySQL database	0x05		
Gwneric Web Server	0x06		
AWS IoT	0x07	○	MQTT receive topic
MS Azure Event hubs	0x08		
Watson IoT for Gateway	0x09	○	MQTT receive topic
MS Azure IoT Hub	0x0a	○	MQTT receive topic
Toami for DOCOMO	0x0b		
UNIX Domain Soket	0x0c		
KDDI IoT cloud Standard	0x0d		
IoT Device Hub	0x0e	○	MQTT receive topic
PD Web	0x0f	○	HTTP response header
Google IoT Core	0x10	○	MQTT receive topic
TCP Soket	0x11	○	Destination IP address and port number.

Cloud ID and the contents of the header

As for Watson IoT, since payload is set as JSON object of "d" key, after obtaining hash value (MD 5), it decodes and outputs only JSON object of "d" key as control message.

For the IoT Device Hub, since the payload is a JSON object of the "parameters" key, after obtaining the hash value (MD 5), it decodes and outputs only the JSON object of the "parameters" key as a control message.

Destination IP address and port number of the TCP socket are added by the following JSON strings.

{“ip_addr”:”<*IP address*>”,”port”:<*port*>}

2.4.PD Web / Plat 'Home proprietary Web Server

2.4.1.HTTP Header

HTTP Header	Description
X-Pd-Web-Version	PD Web specification version number.
X-Pd-Web-Id	Clieant ID
X-Pd-Web-Time	Time stamp conforming to RFC3339
X-Pd-Web-Md5	Hash value(MD5) of payload.
X-Pd-Web-Signature	Hash value created from header information and key.
Content-Type	application/json;charset=UTF-8

2.4.2.Token

The token of the request header(X-Pd-Web-Sinature) created by PD Repeater can be played on the Web server by the following PHP script.

```
$hash_hmac_data =
      $_SERVER['HTTP_X_PD_WEB_VERSION'] .
      $_SERVER['HTTP_X_PD_WEB_ID'] .
      $_SERVER['HTTP_X_PD_WEB_TIME'] .
      $_SERVER['HTTP_X_PD_WEB_MD5'];
$signature = hash_hmac ('sha256', $hash_hmac_data, $key, false);
```

Here, $key is a key that is paired with $_SERVER ['HTTP_X_PD_WEB_ID'] beforehand and saved on the Web server. It authenticates by comparing the played $signature and $_SERVER ['HTTP_X_PD_WEB_SIGNATURE'].

The response header token (X-Pd-Web-Sinature) is created by the following PHP script on the Web server.

```
$tm = localtime();
$timestamp = sprintf("%04d-%02d-%02dT%02d:%02d:%02d.000+09:00",
      $tm[5]+1900,$tm[4]+1,$tm[3],$tm[2],$tm[1],$tm[0]);
$hash_hmac_data = '1.0' . $_SERVER['HTTP_X_PD_WEB_ID'] . $timestamp .
      md5($payload) . $_SERVER['HTTP_X_PD_WEB_SIGNATURE'];
$signature = hash_hmac ('sha256', $hash_hmac_data, $key, false);
```

Here, $payload is the payload strings, if there is no message to send (control message in the downstream direction), $payload = "";

Unlike the request header token, please note that the token of the request header ($_SERVER ['HTTP_X_PD_WEB_SIGNATURE']) sent from the PD Repeater to the signed Strings is included.

3. PD Broker

3.1.Default Path Name

The default path name of files related to PD Broker is as follows.

Path Name	Description
/usr/sbin/pd_broker	Resident execution object (server)
/lib/systemd/system/pd_broker.service	Systemd servicefile
/etc/init.d/pd_broker	RC file
/var/webui/config/pd_broker.conf	Configration file
/var/run/pd_broker.pid	PIDfile

Default path name of files related to PD Broker

3.2.Format of Configuration File

3.2.1.Syntax

```
{
    "brokers" : [
        {
            "enable" : <boolean>,
            "bind": "<UNIX domain socket for input>",
            "buffer_size" : <data buffer size>,
            "destinations":[
                "<UNIX domain socket for output>",
                "<UNIX domain socket for output>"
            ]
        },
        {
            "enable" : <boolean >,
            "bind": "<UNIX domain socket for input>",
            "buffer_size" : <data buffer size>,
            "destinations":[
                "<UNIX domain socket for output>",
                "<UNIX domain socket for output>",
                "<UNIX domain socket for output>"
            ]

        }
    ]
}
```

3.2.2.brokers Object

Key	Data Type	Description
enable	Boolean	Default is false
bind	Strings	The name of the socket that receives data, interpreted as abstract namespace if the beginning of the string is '@'. It is specified in “@ / <*module name*> / <*device number*> .sock” format.
buffer_size	Integer	Data buffer size (byte). Default is 4096.
destinations	Strings array	The destination socket name of the data, interpreted as abstract namespace if the beginning of the string is '@'. It is specified in “@ / <*module name*> / <*device number*> .sock” format. Up to 32 can be specified.

brokers Object

4. PD Agent

4.1.Default Path Name

The default path name of files related to PD Agent is as follows.

Path Name	Description
/usr/sbin/pd_agent	Resident execution object (server)
/lib/systemd/system/pd_agent.service	Systemd servicefile
/etc/init.d/pd_agent	RC file
/var/webui/config/pd_agent.conf	Configration file
/var/run/pd_agent.pid	PIDfile

Default path name of files related to PD Agent

4.2.Format of Configuration File

4.2.1.Syntax

```
{
    "agents" : [
        {
            "enable" : <boolean>,
            “localname:”<device number>”,
            "bind": "<UNIX domain socket for input >",
            “push_to”: “<UNIX domain socket for output>”,
            "buffer_size" : <data buffer size>,
            “channels”:[
                {
                    <channels object>
                },
                {
                    <channels object>
                }
            ]
        },
        {
            "enable" : <boolean>,
            “localname:”<device number>”,
            "bind": "<UNIX domain socket for input>",
            “push_to”: “<UNIX domain socket for output>”,
            "buffer_size" : <data buffer size>,
            “channels”:[
                {
                    <channels object>
                }
            ]
        }
    ]
}
```

4.2.2.agents Object

Key	Data Type	Description
enable	Boolean	Default is false
localname	Strings	Local name of device(Device number). (32byte)
bind	Strings	The name of the socket that receives data, interpreted as abstract namespace if the beginning of the string is '@'. It is specified in “@ / <*module name*> / <*device number*> .sock” format. If empty, default value, @/pd_handler/<*device number*> .sock is specified.
push_to	Strings	The destination socket name of the data, interpreted as abstract namespace if the beginning of the string is '@'. It is specified in “@ / <*module name*> / <*device number*> .sock” format. If empty, default value, @/pd_repeater/<*device number*> .sock is specified.
buffer_size	Integer	Data buffer size (byte). Default is 4096.
channels	JSON array	channels object. Up to 32 can be specified.

agents Object

4.2.3.channels Object

Key	Data Type	Description
name	Strings	Channel name. (32byte)
reply	Boolean	Whether to return response to socket set to ‘push to key’. Default is true
dynamic_link	Boolean	The path name specified in the 'exec' key operates as a Dynamic Link module, default value is false.
index	Strings	JSON string key that evaluated as the start condition of the execution object specified in the 'exec' key. (32byte)
value	Strings	JSON string data value that evaluated as the start condition of the execution object specified in the 'exec' key. (32byte)
exec	Strings	Path name of the execution object to be executed or Dynamic Link module if the JSONStrings data contains the key set to the 'index' key and its value set to the 'value' key. (MAXPATHLEN)
args	Strings	Argument to execute object that is set to 'exec' key. Ignored if 'dynamic_link' key is true. (1024byte)

channels Object

4.3.Environment variables passed to the execution object

If the value is a character string or numeric value, including the key and value used to evaluate the activation condition in the JSON character string of the received data, set it as an environment variable and pass it to the execution object.

Also, the Cloud ID passed from the PD Repeater, the MD5 hash value of the received data (payload), the header information of the received data and the local name of the device set in PD Agent are passed to the environment variable.

Environment Variable Name	Data Type	Description
request_cloud_id	Integer	The number of the cloud to be sent and received assigned for convenience.
request_sub_id	Integer	Identifier when using two servers in the same cloud. (0x00 or 0x01)
request_header	Strings	Header information of received data passed from PD Repeater.
request_payload	Strings	Received data (payload) passed from the PD Repeater.
request_md5	Strings	Hash value (MD5) of received data (payload) passed from PD Repeater.
agent_localname	Strings	Local name of device (device number) that specified in 'localname' key.
agent_bind	Strings	The name of the socket that received the data that specified in 'bind' key.
agent_push_to	Strings	The response destination socket name that specified in 'push to' key.
agent_buffer_size	Integer	Data buffer size that specified in ‘buffer size' key.

Environment variables passed to the execution object except payload

4.4.Dynamic Link module

Here is a sample code of the Dynamic Link module.
The Dynamic Link module is imported into PD Agent by dlopen (). dl_exec () is a function called instead of an external execution object that will be executed by execv ().
The argument to dl_exec () contains all the parameters inherited as environment variables when executing the execution object. If the string pointer result is NULL, the value of the 'result' key of the response message is 'done'. Dl_init () is a function that is called only once when PD Agent is started, and dl_fini () is called only once, please leave sample code unless necessary.

```
/*
 *   Copyright (c) 2018
 *          Plat'Home CO., LTD. <support@plathome.co.jp>. All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *      notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *      notice, this list of conditions and the following disclaimer in the
 *      documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the Plat'Home CO., LTD. nor the names of
```

```
 *      its contributors may be used to endorse or promote products derived
 *      from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
 * WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED.    IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT,
 * INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
 * (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
 * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 */
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <syslog.h>
#include <time.h>
#include <syslog.h>
#include <time.h>
#include <unistd.h>
#include <errno.h>
#include <sys/param.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <sys/un.h>
#include <jansson.h>

#define N_CHANNEL           32

struct channel_t {
        unsigned char     reply;
        unsigned char     dynamic_link;
        char              name[32];                 /* Name of chaanel                                  */
        char              index[32];                /* Index of JSON string to be driven.               */
        char              value[32];                /* Value of JSON string to be driven.               */
        char              exec[MAXPATHLEN];    /* Pathname of dynamic link module.                 */
        int               argc;                     /* Not use in dynamic link module.                  */
        char              *argv[16];                /* Not use in dynamic link module.                  */
        void              *dl_handle;               /* Handle on the global symbol object.              */
};

struct agent_t {
        int               num_ch;                   /* Total number of channels in device.              */
        unsigned char     enable;
        char              localname[32];            /* Local Device name (number).                      */
        char              bind[MAXPATHLEN];    /* Pathname of UNIX domain socket to listen.     */
        size_t            buffer_size;              /* Buffer size of UNIX domain socket.               */
        char              push_to[MAXPATHLEN];    /* Pathname of UNIX domain socket to send. */
        struct channel_t *ch[N_CHANNEL];          /* Pointer to each channel.                         */
        char              *buf;                     /* message buffer for down stream.                  */
};

// Your own functions

/*
        Your own initialize function : void dl_init()
          It is called only once at startup.
          Return value : none.
*/
void dl_init()
```

```
{
        return;
}

/*
        Your own finalize function : void dl_fini()
          It is called only once at termination.
          Return value : none.
*/
void dl_fini()
{
        return;
}

/*
        Your own exec function : int dl_exec()
            char *result                    : Reply value for 'result' key of JSON.
            struct agent_t *agent           : See above structure declaration.
            int ch_number                   : Channel number where Index and Value match.
            json_t *json_obj                : It stores all indexes and values contained in the payload of
                                              the downstream message as Jansson C objects.
            unsigned char cloud_id          : Cloud Id.
            unsigned char sub_id            : Sub Cloud Id.
                                              These are the indications from which cloud they have
                                              been sent.
            char *rcv_header                : Contain the header information of the down stream message.
            char *rcv_payload               : Contain the paylod of the down stream message.
            char *hash                      : Hash value (MD5) of paylod of the down stream message.
          Return value : must be 0 for normal, -1 for error.
*/
int dl_exec(char *result, struct agent_t *agent, int ch_number, json_t *json_obj,
        unsigned char cloud_id, unsigned char sub_id,
        char *rcv_header, char *rcv_payload, char *hash)
{
        int rc;
        char *tx_payload;
        struct channel_t *ch;

        ch = (struct channel_t *)agent->ch[ch_number];

        pid_t pid;

//      The method to obtain Index and value from json_obj is shown below.
/*
        json_t *val;
        const char *key;
        val = json_object();
        json_object_foreach(json_obj, key, val) {
                if(json_is_string(val)) {
                        printf("%s", json_string_value(val));
                }
                else if(json_is_integer(val)) {
                        printf("%ld", json_integer_value(val));
                }
                else if(json_is_real(val)) {
                        printf("%lf", json_real_value(val));
                }
        }
*/

//      Write your own process here.

        snprintf(result, sizeof(char)*agent->buffer_size,
```

```
                "{YOUR OWN JSON OBJECT}");

        return(0);
}
```

4.5.Response Message

If the 'reply' key is true, PD Agent returns the execution status as a JSON string to the socket set to 'push to'.

Status	Response Message
Started an execute object	{"time":"<*timestamp*>","repl_ to":"<*md5*>","result":"queuing", "reason":"matched","matched":{"<*key*>":"<*value*>"}}
Failed to start an execute object	{"time":"<*timestamp*>","repl_ to":"<*md5*>","result":"faild", "reason":"matched","matched":{"<key>":"<value>"}}
Finished to execute object	{"time":"<*timestamp*>","repl_ to":"<md5>","result":"done", "reason":"matched","matched":{"<*key*>":"<*value*>"}}
Matching keys and values not exist	{"time":"<*timestamp*>","repl_ to":"<*md5*>","result":"not queuing", "reason":"key and value notmatched"}
Received data is not in JSON format	{"time":"<*timestamp*>","repl_ to":"<*md5*>","result":"not queuing", "reason":"not JSON form"}

PD Agent response message

Here, <*md 5*> is the MD5 hash value of the received data passed from the pd repeater, " <*key*> “: "<*value*>"' matches the matching 'index' key and 'value’ key, It is a set condition.

Since the execution status is the return value of execv () which calls the execution object, it does not indicate the result of the processing of the execution object. Therefore, if the integrity of the system is to be obtained, a response is returned to a predetermined Unix domain socket by an execution object Please do it.
When using the Dynamic Link module and the character string pointer argument result of dl_exec () is not NULL, the value of the 'result' key at the end of execution is replaced with the value of the string pointer argument result (JSON string).

5.PD Handler Modbus

5.1.Modbus Function Code

Code	Function Name	Description
0x01	Read Coils	Read the bit value set for digital output.
0x02	Read Discrete Input	Read bit value of digital input.
0x03	Read Holding Registers	Read the value set in the register output.
0x04	Read Input Registers	Read the register input value.
0x05	Write Single Coil	Write the bit value to digital output 1 bit.
0x06	Write Single Register	Write the value to register output 1 register.
0x07	Read Exception Status	Notify error status between Modbus server / client.
0x09	Write Single Discrete Input	Write the bit value to digital input 1 bits.
0x0a	Write Single Input Register	Write the value to register input 1 register.
0x0f	Write Multiple Coils	Write bit values to multiple consecutive digital outputs.
0x10	Write Multiple Registers	Write values to multiple consecutive register outputs.
0x11	Report Slave ID	A list of unit IDs of connectable slave (server) devices.
0x13	Write Multiple Discrete Input	Write bit values to multiple consecutive digital inputs.
0x14	Write Multiple Input Registers	Write values to multiple consecutive register inputs.
0x16	Mask Write Registers	Mask register output.
0x17	Write And Read Registers	Write values to multiple consecutive register outputs and read their values.

Modbus Function Code for PD Handler

00x09, 0x0a, 0x13, 0x14 are proprietary specifications different from the original Modbus protocol prepared to be able to set digital input or register input of PD Handler Modbus Server without physical input from the cloud side.

The Modbus function code is set to the value of the 'function' key of the JSON string in the setting file or the CSV file or control from the cloud side.
It is also possible to set the function code in hexadecimal notation starting from '0x' to the value of 'function' key as Strings, but it is also possible to use integer notation and string notation in addition to hexadecimal notation.

Code	Function Name	Integer Notation	Strings Notation
0x01	Read Coils	1	read_coils
0x02	Read Discrete Input	2	read_discrete_input
0x03	Read Holding Registers	3	read_holding_registers
0x04	Read Input Registers	4	read_input_registers
0x05	Write Single Coil	5	write_single_coil
0x06	Write Single Register	6	write_single_register
0x07	Read Exception Status	7	read_exception_status
0x09	Write Single Discrete Input	9	write_single_discrete_input
0x0a	Write Single Input Register	10	write_single_input_register
0x0f	Write Multiple Coils	15	write_multiple_coils
0x10	Write Multiple Registers	16	write_multiple_registers
0x11	Report Slave ID	17	report_slave_id
0x13	Write Multiple Discrete Input	19	write_multiple_discrete_input
0x14	Write Multiple Input Registers	20	write_multiple_iput_registers
0x16	Mask Write Registers	22	mark_write_registers
0x17	Write And Read Registers	23	write_and_read_registers

Alternate name for function code used for 'function' key

5.2.PD Handler Modbus Client

5.2.1.Default Path Name

The default path name of files related to PD Handler Modbus Client is as follows.

Path Name	Description
/usr/sbin/pd_handler_modbus_client	Resident execution object (server)
/lib/systemd/system/pd_handler_modbus_client.service	Systemd servicefile
/etc/init.d/pd_handler_modbus_client	RC file
/var/webui/config/pd_handler_modbus_client.conf	Configration file
/var/webui/upload_dir/ pd_handler_modbus_client.csv	CSV file
/var/run/pd_handler_modbus_client.pid	PIDfile

Default path name of files related to PD Handler Modbus Client

5.2.2.Format of Configuration File

5.2.2.1.Syntax

```
{
    "csv_file" : “<path name of CSV file>”,
    “clients”: [
        {
            <clients object>,
            “acquisitions”:[
                {
                    <acquisitions object>
                },
                {
                    <acquisitions object>
                }
            ]
        },
        {
            <clients object>,
            “acquisitions”:[
                {
                    <acquisitions object>
                },
                {
                    <acquisitions object>
                }
            ]
        }
    ]
}
```

5.2.2.2.Root Object

Key	Data Type	Description
csv_file	Strings	Path name of CSV file. Default is '/var/webui/upload_dir/pd-data/pd_handler_modbus_client.csv'. (MAXPATHLEN)
clients	JSON obj	clients object

Root Object

5.2.2.3.clients Object

clients object has up to 256 arrays.

There are common objects independent of Modbus protocol (TCP, RTC) dependent objects and protocols.

Key	Protocol	Data Type	Description
enable	Common	Boolean	Default is false
localname		Strings	Local name of device(Device number). (32byte)
memo		Strings	A user-defined string appended to the JSON character output as the value of the 'memo' key. (256byte)
bind		Strings	The name of the socket that receives data, interpreted as abstract namespace if the beginning of the string is '@'. It is specified in “@ / <*module name*> / <*device number*> .sock” format. If empty, default value, @/pd_handler/<*device number*> .sock is specified.
push_to		Strings	The destination socket name of the data, interpreted as abstract namespace if the beginning of the string is '@'. It is specified in “@ / <*module name*> / <*device number*> .sock” format. If empty, default value, @/pd_repeater/<*device number*> .sock is specified.
buffer_size		Integer	Data buffer size (byte). Default is 4096.
receive		Boolean	Whether to receive a message from the cloud (downstream direction control) or not. Default is false
protocol		Strings	Connection protocol,specify 'tcp' or 'rtu' . Default is 'tcp'.
writeout		Boolean	Whether to output to socket specified 'push to'. Default is true.
interval		Integer	The interval (sec) for acquiring data.(sec) Default is 60.
timeout		Integer	Timeout when data can not be acquired (msec). Default is 5000.
time_sysnc		Boolean	Time synchronize mode. Default is false
base_time		Strings	Specify the reference time in the time synchronize mode in 'HH:MM' format. Default is '00:00'.
node	TCP	Strings	IP address of TCP connected PLC device. Default '127.0.0.1'
port		Integer	Port number of TCP connected PLC device . Default is 502
device	RTU	Strings	Device name of serial connected PLC device. Default is '/dev/tty00'
rtu_speed		Integer	Baud rate of serial connected PLC device. Default is 115200
rtu_bits		Integer	Number of bits of serial connected PLC device, 7 or 8. Default is 8.
rtu_parity		Strings	Parity of serial connected PLC device, Either ‘none','even','odd'. Default is 'none'.
rtu_stop		Integer	Stop bit of serial connected PLC device, 1 or 2. Default is 1.
acquisitions	Common	JSON array	acquisitions object

clients Object

5.2.2.4.acquisitions Object

Key	Data Type	Description
unit	Integer	Modbus unit ID of PLC device. 1 to247 or 255 for TCP.
function	Strings or Integer	Data read function name or function number of Modbus protocol. In the case of strings, either 'read_coils', 'read_discrete input', 'read_holding_registers','read_input_registers' or '0x01' to '0x04'. In the case of integer, in order of strings notation 1 to 4. Default is 'read_holding_r_gisters'.
data_type	Strings or Integer	In the case of strings, either 'uint16_t', 'int16_t', 'uint32lsb_t', 'int32lsb_t', 'uint32msb_t','int32msb_t' . In the case of integer , in order of strings notation 0 to 5. If value of ‘function’ key is 'read_coils' or 'read discrete_input', it is fixed to 'uint16_t'. Defult is 'uint16 t'.
address	Strings or Integer	Specify the start address of the register to be read. If '0x' is appended to the beginning of a character string, it is interpreted as hexadecimal notation. Default is '0x0'.
number	Strings or Integer	Number of bits or registers to be read . If '0x' is appended to the beginning of a character string, it is interpreted as hexadecimal notation. If the data_type key value is 'uint32lsb_t', 'int32ls_t', 'uint32msb_t', 'int32msb_t', internally it will be treated as twice the value.The default value is 1.

acquisitions Object

5.2.2.5.Available Function Codes

Modbus function code that available on PD Handler Modbus Client

Code	Function Name	On Local	From Cloud
0x01	Read Coils	○	○
0x02	Read Discrete Input	○	○
0x03	Read Holding Registers	○	○
0x04	Read Input Registers	○	○
0x05	Write Single Coil		○
0x06	Write Single Register		○
0x07	Read Exception Status		
0x09	Write Single Discrete Input		
0x0a	Write Single Input Register		
0x0f	Write Multiple Coils		○
0x10	Write Multiple Registers		○
0x11	Report Slave ID		○
0x13	Write Multiple Discrete Input		
0x14	Write Multiple Input Registers		
0x16	Mask Write Registers		
0x17	Write And Read Registers		○

Modbus function code that available on PD Handler Modbus Client

“On Local” means a function code that can be specified in the configuration file or CSV file, “From Clou”' is intended as a function code that can be specified in downstream messaage from the cloud.

5.2.2.6.CSV File

By setting the CSV file that in the value of csv_file key of the configuration file (/var/webui/config/pd_handler_modbus_client.conf), It can be set multiple acquisition for one device number set in the configuration file.

The format of the CSV file is as follows.

Device number, Unit ID, Reading method, Data type, Read address, Number of read

Parameta	Data type	Description
Device number	ASCII	Specify the device number specified in the localname key of the Clients object, Undefined device numbers are ignored. If the first character is '#' or '/', it is treated as a comment line.
Unitid ID	Alphameric	Modbus unit ID of PLC device. 1 to247 or 255 for TCP.
Reading method	ASCII	Specify one of the following as the reading method. Digital output：'read_coils' or '0x01' or '1' Digital input：'read_discrete_input' or '0x02' or '2' Register output：'read_holding_registers' or '0x03' or '3' Register input：'read_input_registers' or '0x04' or '4'
Data type	ASCII	Specify one of the following as the data type. Unsigned 16bits：'uint16_t' or '0' Signed 16bits：'int16_t' or '1' Unsigned 32bits LSB：'uint32lsb_t' or '2' Signed 32bits LSB：'int32lsb_t' or '3' Unsigned 32bits MSB：'uint32msb_t' or '4' Signed 32bits MSB：'int32msb_t' or '5'
Read address	ASCII	Specify the start address of the register to be read. If '0x' is appended to the beginning of a character string, it is interpreted as hexadecimal notation. Default is '0x0'.
Number of read	Alphameric	Number of bits or registers to be read . If '0x' is appended to the beginning of a character string, it is interpreted as hexadecimal notation. If the data_type key value is 'uint32lsb_t', 'int32ls_t', 'uint32msb_t', 'int32msb_t', internally it will be treated as twice the value.The default value is 1.

CSV file format of PD Handler Modbus Client

Each parameter is delimited by a comma ',', and the first sharp '#' or slash '/' is regarded as a comment line.

When the CSV file is loaded, the Modbus client device specified in Clients object will be overwritten. Therefore, in the CSV file, please describe all Modbus client devices including Modbus client device set with Clients object.

5.2.2.7.Reference Time Control

Reference time control is a function to acquire data at a specific time.
Specify the ‘time_sysnc’ key of Clients object to ‘true’, and set the acquisition interval and time with the ‘interval’ key and ‘base_time’ key.

"Acquisition time interval" in the reference time control is limited to multiples of 300, 600, 900, 1800, 3600, 7200, 10800, 14400, 21600, 28800, 43200 and 86400. If values other than these are set as "Acquisition time interval", it is handled as follows in the PD Handler Modbus Client.

Time interval setting	Actual operation
0 to 599	300
600 to 899	600
900 to 1799	900
1800 to 3599	1800
3600 to 7199	3600
7200 to 10799	7200
10800 to 14399	10800
14400 to 21599	14400
21600 to 28799	21600
28800 to 43199	28800
43200 to 86399	43200
86400 or more	Multiples of 86400

Setting of acquisition time interval and actual operation

"Reference time" is the time at which the operation starts, for example, when "acquisition time interval" is set to 300 and "reference time" is set to "00:01", the acquisition of data is 00:01, 00:06, 00: 11 ... 00:56, 01:01 ... 23:56, 00:01 on schedule.
The acquisition start time of data is not the time itself set as "reference time" but the latest time calculated from "reference time" and "acquisition time interval".
For example, when "reference time" "01: 05" and "acquisition time interval" 10800 are set at 08:30, the first data acquisition is performed at 10:05, and thereafter at 13:05, 16:05, It will be done in order of 19: 05, 22: 05, 01: 05.

5.3.PD Handler Modbus Server

The Modbus server holds the register map shown in the following table, waits for connection by the Modbus protocol from the PLC device, updates the register value by the write operation of the PLC device, updates the updated register and its value via the PD Repeater will send it to the cloud.

It can also read and write a register map based on the downstream message (JSONS character string) sent from the cloud via the PD Repeater.

The register map is output to the register map file registers.map if there is an update every 60 seconds.

Register	Start Address	Size
Digital Output (Coils)	0x000	uint8_t × 2048
Digital Input(Discrete Input)	0x000	uint8_t × 2048
Register Output(Holdig Registers)	0x000	uint16_t × 2048
Register Input(Input Registers)	0x000	uint16_t × 2048

Registers Map of PD Handler Modbus Server

5.3.1.Default Path Name

The default path name of files related to PD Handler Modbus Server is as follows.

Path Name	Description
/usr/sbin/pd_handler_modbus_server	Resident execution object (server)
/lib/systemd/system/pd_handler_modbus_server.service	Systemd servicefile
/etc/init.d/pd_handler_modbus_server	RC file
/var/webui/config/pd_handler_modbus_server.conf	Configration file
/var/webui/.modbus_server/registers.map	Registers map file
/var/run/pd_handler_modbus_server.pid	PID file

Default path name of files related to PD Handler Modbus Server

5.3.2.Format of Configuration File

5.3.2.1.Syntax

```
{
    "registers_file" : "<path name of registers map file>",
    "servers": [
        {
            <servers object>
        },
            <servers objects>
        }
    ]
}
```

5.3.2.2.Root Object

Key	Data Type	Description
registers_file	Strings	Path name of registers map file. Default is '/var/webui/.modbus_server/pd-data/registers.map'. (MAXPATHLEN)
servers	JSON obj	aervers object

Root Object

5.3.2.3.servers Object

The server object is a setting object that regulates the operation of PD Handler Modbus Server.

There are up to eight Array of server objects, there are common objects independent of Modbus protocol (TCP, RTC) dependent objects and protocols.

Key	Protocol	Data Type	Description
enable	Common	Boolean	Default is false
localname		Strings	Local name of device(Device number). (32byte)
memo		Strings	A user-defined string appended to the JSON character output as the value of the 'memo' key. (256byte)
bind		Strings	The name of the socket that receives data, interpreted as abstract namespace if the beginning of the string is '@'. It is specified in "@ / <*module name*> / <*device number*> .sock" format. If empty, default value, @/pd_handler/<*device number*> .sock is specified.
push_to		Strings	The destination socket name of the data, interpreted as abstract namespace if the beginning of the string is '@'. It is specified in "@ / <*module name*> / <*device number*> .sock" format. If empty, default value, @/pd_repeater/<*device number*> .sock is specified.
buffer_size		Integer	Data buffer size (byte). Default is 4096.
receive		Boolean	Whether to receive a message from the cloud (downstream direction control) or not. Default is false
protocol		Strings	Connection protocol,specify 'tcp' or 'rtu' . Default is 'tcp'.
writeout		Boolean	Whether to output to socket specified 'push to'. Default is true.
timeout		Integer	Timeout when data can not be acquired (msec). Default is 5000.
node	TCP	Strings	Listen IP address of TCP connection. Default '127.0.0.1'
port		Integer	Listen port number of TCP connection . Default is 502
device	RTU	Strings	Device name of serial connection. Default is '/dev/tty00'
rtu_speed		Integer	Baud rate of serial connection. Default is 115200
rtu_bits		Integer	Number of bits of serial connection, 7 or 8. Default is 8.
rtu_parity		Strings	Parity of serial connection, Either 'none','even','odd'. Default is 'none'.
rtu_stop		Integer	Stop bit of serial connection, 1 or 2. Default is 1.
unit		Integer	Modbus Unit ID to be given to itself. 1 ～ 247.

servers Object

5.3.2.4.Available Function Codes

Modbus function code that available on PD Handler Modbus Server.

Code	Function Name	On Local	From Cloud
0x01	Read Coils	○	○
0x02	Read Discrete Input	○	○
0x03	Read Holding Registers	○	○
0x04	Read Input Registers	○	○
0x05	Write Single Coil	○	○
0x06	Write Single Register	○	○
0x07	Read Exception Status	○	
0x09	Write Single Discrete Input		○
0x0a	Write Single Input Register		○
0x0f	Write Multiple Coils	○	○
0x10	Write Multiple Registers	○	○
0x11	Report Slave ID	○	○
0x13	Write Multiple Discrete Input		○
0x14	Write Multiple Input Registers		○
0x16	Mask Write Registers	○	
0x17	Write And Read Registers	○	○

Modbus function code that available on PD Handler Modbus Server

“On Local” means a function code that can be specified in the configuration, “From Clou”' is intended as a function code that can be specified in downstream messaage from the cloud.

6.PD Handler BLE (Node.js)

6.1.Default Path Name

The default path name of files related to PD Handler BLE (Node.js).

Path Name	Description
/opt/pd/handler/pd-handler.js	Execution object
/etc/init.d/pd-handler-ble-js	RC file
/var/webui/config/pd-handler-ble.conf	Configration file
/var/run/pd-handler-ble-js.pid	PID file
/var/webui/pd-logs/pd-handler-ble-js.log	Log file
/var/werbui/.blebackup/pd-handler-ble.conf	Backup of configration file
/var/werbui/.blebackup/pd-handler-ble-js-restore.log	File for restore
/var/webui/.bsensor/xxx.json	Data file for local viewer.(xxx is localname.)

Default path name of files related to PD Handler BLE (Node.js)

6.2.Format of Configuration File

6.2.1.Syntax

{

“servers”: <*servers object*>,

“beacon”: <*beacon object*>,

“blesensor”: <*blesensor object*>

}

6.2.2.Root Object

Key	Data Type	Description
servers	JSON obj	servers object
beacon	JSON obj	beacon object
blesensor	JSON array	blesensor object

Root Object

6.2.3.servers Object

Key	Data Type	Description
json_ldir	Strings	Directory path name for data.
handler_ldir	Strings	Directory path name for backup.
blehandler_monitor_api	Strings	API file for beacon monitoring.

serevers Object

6.2.4.beacon Object

Key	Data Type	Description
localname	Strings	Local name of device(Device number). (32byte)
push_to	Strings	The destination socket name of the data, interpreted as abstract namespace if the beginning of the string is '@'. It is specified in “@ / <*module name*> / <*device number*> .sock” format. If empty, default value, @/pd_repeater/<*device number*> .sock is specified.
duplicate_type	Strings	“interval”, “entry”, “inout”
duplicate_interval	Integer	Overlap control time interval. 0 to 3600000[msec]
appendix_info	Strings	Appendix information. Default is serial number
payload_manage	JSON obj	payload_manage object.
data_filter_rule	JSON array	data_filter_rule object.
rssi_filter	Integer	RSSI filter
add_data	JSON obj	User definition information.
enable	Boolean	Whether or not to transmit data.

beacon Object

6.2.4.1.payload_manage Object

Key	Data Type	Description
data	Boolean	Whether or not to add advertisement data to transmission data.
localname	Boolean	Whether or not to give a local name to transmission data.
type	Boolean	Whether or not to add a beacon type (iBeacon) to transmission data.

payload_manage Object

6.2.4.2.data_filter_rule Object

Key	Data Type	Description
length	Integer	Length of prefix.
prefix	Strings	Forward-matched data filter.

data_filter_rule Object

6.2.5.blesensor Object

Key	Data Type	Description
localname	Strings	Local name of device(Device number). (32byte)
push_to	Strings	The destination socket name of the data, interpreted as abstract namespace if the beginning of the string is '@'. It is specified in “@ / <*module name*> / <*device number*> .sock” format. If empty, default value, @/pd_repeater/<*device number*> .sock is specified.
uuid	Strings	Bluetooth Device Address (Colon deleted, hexadecimal number)
memo	Strings	User defined strings appended to the output.. (256byte)
interval	Integer	Data transmission interval.[msec]
txpower	Integer	Transmission power [dBm]
local	Boolean	Whether to transmit to the local viewer.
enable	Boolean	Whether or not to transmit data.

blesensor Object

7.PD Handler BLE (C)

7.1.Default Path Name

The default path name of files related to PD Handler BLE (C).

Path Name	Description
/usr/sbin/pd-handler-ble-c	Resident execution object (server)
/etc/init.d/pd-handler-ble-c	RC file.
/var/webui/config/pd-handler-ble.conf	Configration file.
/var/run/pd-handler-ble-c.pid	PID file.
/var/webui/pd-logs/pd-handler-ble-c.log	Log file.
/var/werbui/.blebackup/pd-handler-ble.conf	Backup of configration file.
/var/werbui/.blebackup/pd-handler-ble-c-restore.log	File for restore.
/var/webui/.bsensor/xxx.json	Data file for local viewer.(xxx is localname.)
/opt/pd/lua/ble/devices/*.lua	Lua extension file.

Default path name of files related to PD Handler BLE (C)

7.2.Format of Configuration File

The configuration file is the same as PD Handler BLE (Node.js), see section 6.2.

8.PD Handler UART

8.1.Default Path Name

The default path name of files related to PD Handler UART.

Path Name	Description
/usr/sbin/pd-handler-uart	Resident execution object (server)
/etc/init.d/pd-handler-uart	RC file.
/var/webui/config/pd-handler-uart.conf	Configration file.
/var/run/pd-handler-uart.pid	PID file.
/var/webui/pd-logs/pd-handler-uart.log	Log file.
/opt/pd/lua/uart/enocean/devices/*.lua	Lua extension file.

Default path name of files related to PD Handler UART

8.2.Format of Configuration File

8.2.1.EnOcean

8.2.1.1.Syntax

{

“prototype”:”enocean”,

“enocean”: <*enocean object*>,

“enocean_device”: <*enocean_device object*>

}

8.2.1.2.Root Object

Key	Data Type	Description
prototype	Strings	“enocean”
enocean	JSON obj	enocean object
enocean_device	JSON array	enocean_device object

Root Object

8.2.1.3.enocean Object

Key	Data Type	Description
serial_port	Strings	Default is ”/dev/ttyEX2”
raw_data_mode	Boolean	If true, send received data as is

enocean Object

8.2.1.4.enocean_device Object

Key	Data Type	Description
localname	Strings	Local name of device(Device number). (32byte)
push_to	Strings	The destination socket name of the data, interpreted as abstract namespace if the beginning of the string is '@'. It is specified in “@ / <*module name*> / <*device number*> .sock” format. If empty, default value, @/pd_repeater/<*device number*> .sock is specified.
deviceid	Strings	ID
memo	Strings	User defined strings appended to the output.. (256byte)
eep	Strings	EnOcean Equipment Profiles
enable	Boolean	Whether or not to transmit data.

enocean_device Object

Plat'Home Co., Ltd.
NIHON BUILDING KUDANBEKKAN,3F
4-2-3,Kudankita,Chiyoda-ku,TOKYO 102-0073 JAPAN