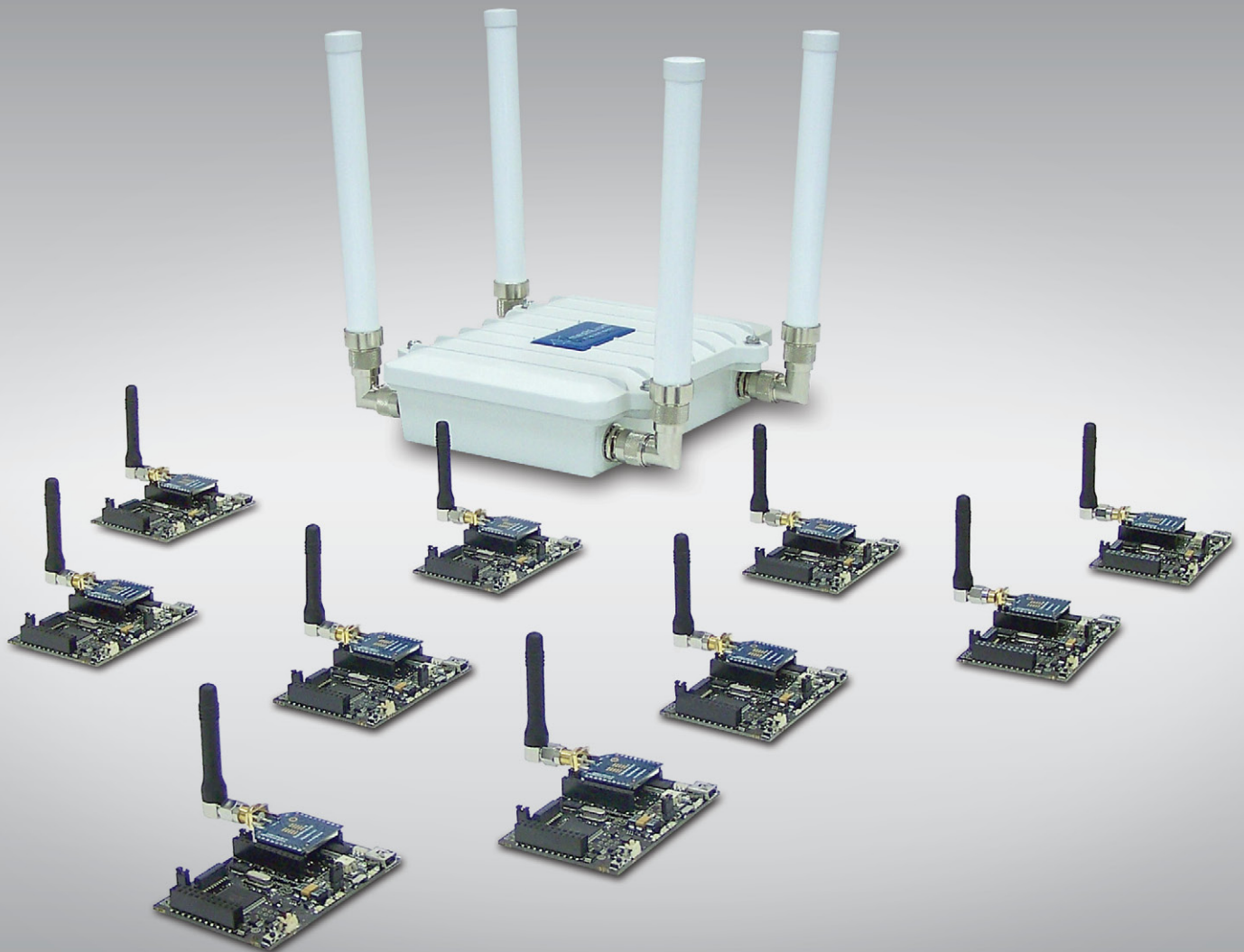


Wireless Sensor Networks with Wasp mote and Meshlium



Wireless Sensor Networks with Wasp mote and Meshlium

- **Three Libelium technologies:**

Wasp mote is a sensor device specially oriented to developers. It works with different protocols (ZigBee, Bluetooth, GPRS) and frequencies (2.4GHz, 868MHz, 900MHz) being capable of getting links up to 12km. It counts with an hibernate mode of 0.7uA which allows to save battery when it is not transmitting. More than 50 sensors already available and a complete open source IDE (API libraries + compiler) made really easy to start working with the platform.



The new Wasp mote Plug & Sense! line allows developers to forget about electronics and focus on services and applications. Now you can deploy wireless sensor networks in a easy and scalable way ensuring minimum maintenance costs. The new platform consists of a robust waterproof enclosure with specific external sockets to connect the sensors, the solar panel, the antenna and even the USB cable in order to reprogram the node. It has been specially designed to be scalable, easy to deploy and maintain.



More info: <http://www.libelium.com/waspote>

Meshlium is a Linux router which works as the Gateway of the Wasp mote Sensor Networks. It can contain 5 different radio interfaces: Wifi 2.4GHz, Wifi 5GHz, 3G/GPRS, Bluetooth and **ZigBee**. As well as this, Meshlium can also integrate a GPS module for mobile and vehicular applications and be solar and battery powered. These features a long with an aluminium IP-65 enclosure allows Meshlium to be placed anywhere outdoor. Meshlium comes with the Manager System, a web application which allows to control quickly and easily the Wifi, ZigBee, Bluetooth and 3G/GPRS configurations a long with the storage options of the sensor data received.



The new Meshlium Xtreme allows to detect iPhone and Android devices and in general any device which works with Wifi or Bluetooth interfaces. The idea is to be able to measure the amount of people and cars which are present in a certain point at a specific time, allowing the study of the evolution of the traffic congestion of pedestrians and vehicles.

More info: <http://www.libelium.com/meshlium>

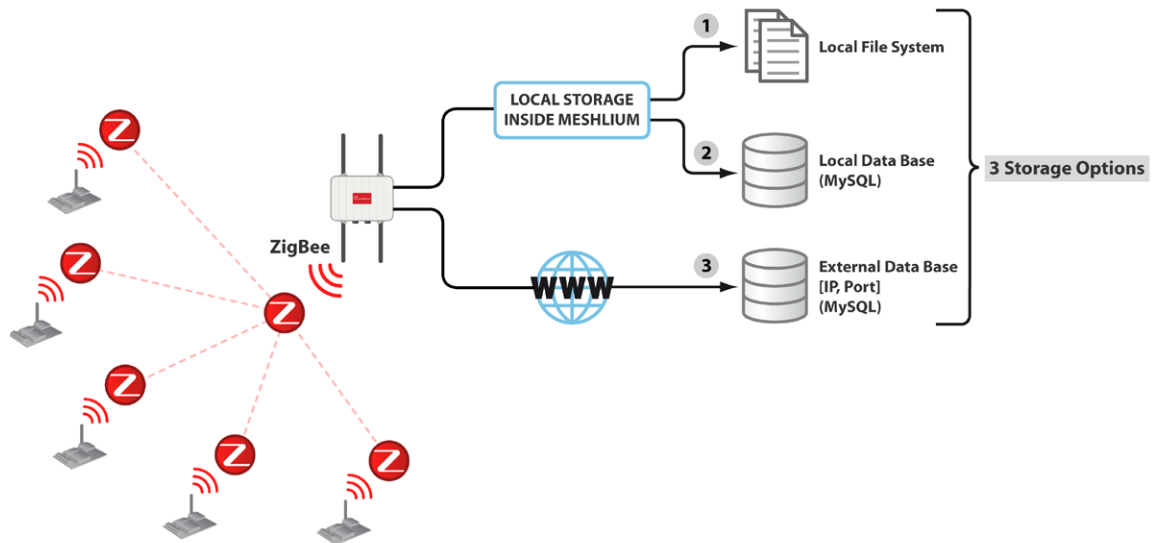
How do they work together?

Meshlium receives the sensor data sent by Wasp mote using the ZigBee radio.

Then 5 possible actions can be performed:

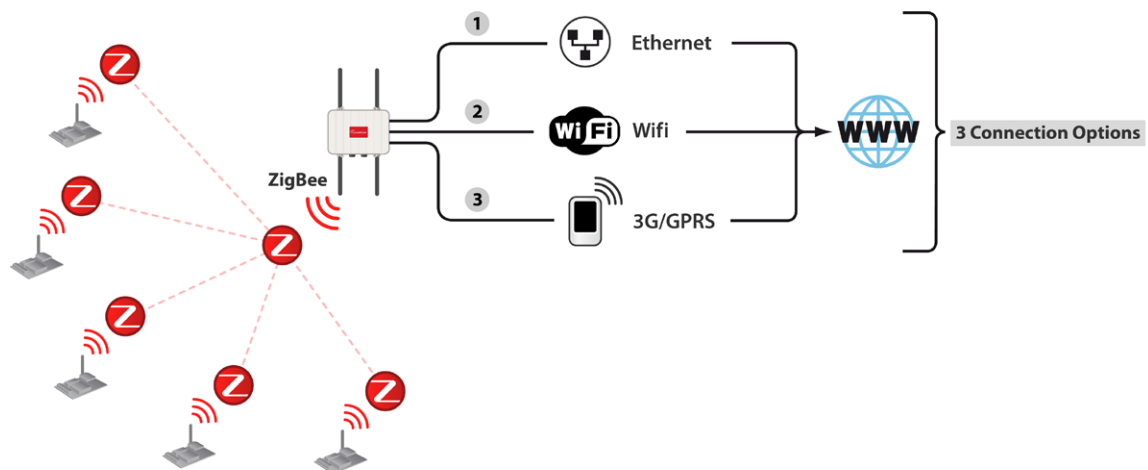
1. Store the sensor data in the Meshlium file system
2. Store the sensor data in the Meshlium Local Data Base (MySQL)
3. Store the ZigBee sensor data in an External Data Base (MySQL)
4. Send the information to the Internet using the Ethernet or Wifi connection
5. Send the information to the Internet using the 3G/GPRS connection

• Meshlium Storage Options



- Local File System
- Local Data Base
- External Data Base

• Meshlium Connection Options



- ZigBee -> Ethernet
- ZigBee -> Wifi
- ZigBee -> 3G/GPRS

• Capturing and storing sensor data in Meshlium from a Waspote sensor network

When you buy a kit containing Meshlium and Waspote, they already come configured to work together. Meshlium will receive the sensor data sent by Waspote using the ZigBee radio and it will store in the Local File System and in the Local Data Base.

The initial ZigBee frames sent by Waspote contain the next sequence:

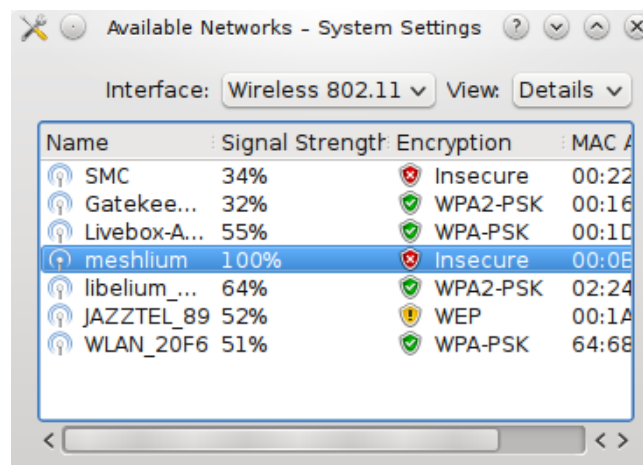
```
[HEADER] -mac:0013a20040307f9c -x:27,y:23,z:1023 -temp:28 -bat:97%
```

They are formed by the MAC address (64b), the acceleration in the three axis (x, y, z) , the temperature and the battery level.

Meshlium comes with all the radios ready to be used. Just "plug & mesh!". All the Meshlium nodes come with the Wifi AP ready so that users can connect using their Wifi devices. Connect the ethernet cable to your network hub, restart Meshlium and it will automatically get an IP from your network using DHCP *.

(*) For the Meshlium Mesh AP and for the Meshlium ZigBee Mesh AP the Internet connection depends on the GW of the network.

Then access Meshlium through the Wifi connection. First of all search the available access points and connect to "Meshlium".



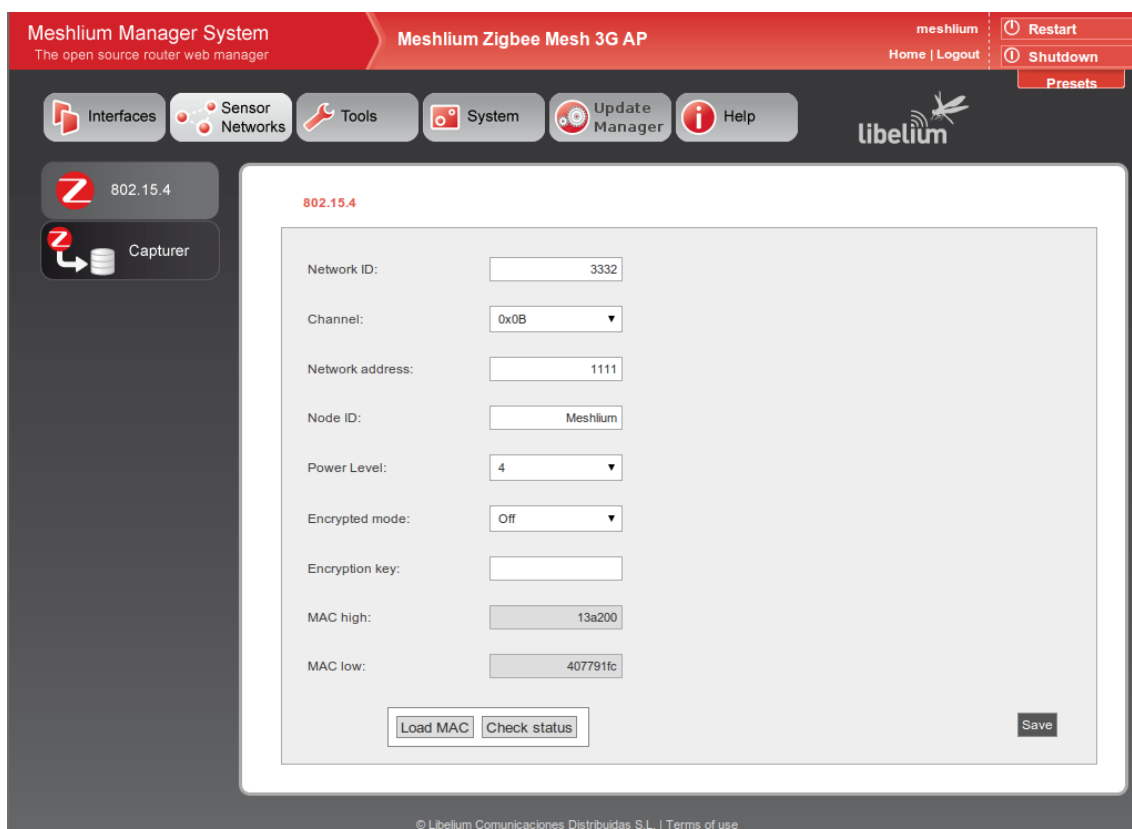
No password is needed as the network is public (you can change it later in the Wifi AP Interface options). When you select it, Meshlium will give an IP from the range 10.10.10.10 - 10.10.10.250.

Now you can open your browser and access to the Meshlium Manager System:

- **URL:** http://10.10.10.1/ManagerSystem
- **user:** root
- **password:** libelium



Now we go to the "Sensor Networks" tab.



There are 6 different XBee models can can be configured:



Depending the kind of XBee model the parameters to be configured may vary.

Complete list:

- **Network ID:** Also known as PAN ID (Personal Area Network ID)
- **Channel:** frequency channel used
- **Network Address:** 16b address (hex field) - MY
- **Node ID:** maximum 20 characters (by default "Meshlium")
- **Power level:** [0..4] (by default 4)
- **Encrypted mode:** true/false (by default false)
- **Encryption Key:** 16 characters maximum
- **MAC:** 64b hardware address. It is a read only value divided in two parts:
 - MAC-high: 32b (hex field)
 - MAC-low: 32b (hex field)

These parameters must be also configured in the WaspMote sensor nodes. Access to all the information related to WaspMote at: <http://www.libelium.com/waspmote>

DigiMesh

Network ID:	<input type="text" value="3332"/>
Channel:	<input type="text" value="0x0E"/>
Node ID:	<input type="text" value="Meshlium"/>
Power Level:	<input type="text" value="2"/>
Encrypted mode:	<input type="text" value="Off"/>
Encryption key:	<input type="text"/>
MAC high:	<input type="text" value="13a200"/>
MAC low:	<input type="text" value="407791fc"/>

To discover the MAC address of the XBee module just press the “Load MAC” button.

The “Check status” option allows to see if the ZigBee radio is working properly and if the configuration stored on it matches the values set in the Manager System.

Both process ("Load MAC" and "Check status") require the ZigBee capturer daemon to be stopped. This means no frames will be received while executing this actions. Be patient this can take up to 1 minute to finish.

DigiMesh

Network ID:	<input type="text" value="3332"/>	<div>Connecting to serial port ... Connected.</div> <div> Network ID: OK Node ID: OK Power Level: OK Encrypted Mode: OK </div>
Channel:	<input type="text" value="0x0E"/>	
Node ID:	<input type="text" value="meshlium"/>	
Power Level:	<input type="text" value="2"/>	
Encrypted mode:	<input type="text" value="Off"/>	
Encryption key:	<input type="text"/>	
MAC high:	<input type="text" value="13a200"/>	
MAC low:	<input type="text" value="407791fc"/>	

Note: When you buy a WaspMote Developer kit with Meshlium and with the XBee ZB as ZigBee radio both the WaspMote GW and Meshlium come configured as Coordinator of the network. Take into account that only one of them can be working at the same time.

Note: If the encryption check fails but the rest of parameters are OK, it means the ZigBee radio has an old version of the firmware but it is working perfectly.

• Capturing and storing sensor data

When you buy a kit containing Meshlium and Waspote, they already come configured to work together. Meshlium will receive the sensor data sent by Waspote using the ZigBee radio and it will store in the Local File System and in the Local Data Base.

The initial ZigBee frames sent by Waspote contain the next sequence:

```
[HEADER] -mac:0013a20040307f9c -x:27,y:23,z:1023 -temp:28 -bat:97%
```

They are formed by the MAC address (64b), the acceleration in the three axis (x, y, z), the temperature and the battery level.

In order to add your own sensor frames properly go to the section *"Capturing and storing your own ZigBee frames"*. However if you change the sensor information sent by Waspote without changing the default capturer daemon it will be saved as a generic "Frame" in the database. See the picture below in order to see different frames types and how they are saved in the database.

Default Waspote frames



ID	Timestamp	Frame
410	2012-05-03 12:58:50	MAC: 0013a200407791fc - X: -28 - Y: 47 - Z: 1056 - Temperature: 30 - Battery: 98%
409	2012-05-03 12:58:49	MAC: 0013a200407791fc - X: -26 - Y: 48 - Z: 1057 - Temperature: 30 - Battery: 98%
408	2012-05-03 12:58:47	MAC: 0013a200407791fc - X: -27 - Y: 47 - Z: 1057 - Temperature: 30 - Battery: 98%
407	2012-05-03 12:58:46	MAC: 0013a200407791fc - X: -27 - Y: 47 - Z: 1057 - Temperature: 30 - Battery: 98%
406	2012-05-03 12:58:45	MAC: 0013a200407791fc - X: -28 - Y: 48 - Z: 1057 - Temperature: 30 - Battery: 98%
405	2012-05-03 12:54:41	Frame: r#@w gps position: 41.680589,-0.886454 radiation: 0.041usv
404	2012-05-03 12:54:40	Frame: r#@w gps position: 41.680589,-0.886454 radiation: 0.041usv
403	2012-05-03 12:54:38	Frame: r#@w gps position: 41.680589,-0.886454 radiation: 0.041usv
402	2012-05-03 12:54:37	Frame: r#@w gps position: 41.680589,-0.886454 radiation: 0.041usv

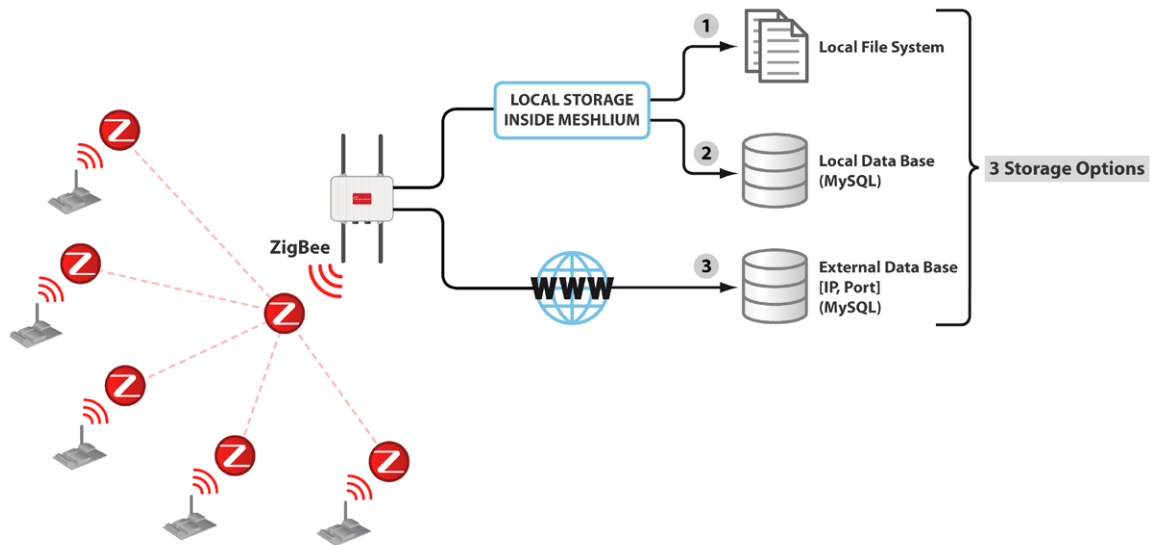
New ZigBee frames from user
802.15.4 Header
Sensor values

In order to work with new sensor information added to the ZigBee frames go to the *"Capturing and Storing new sensor data frames"* chapter.

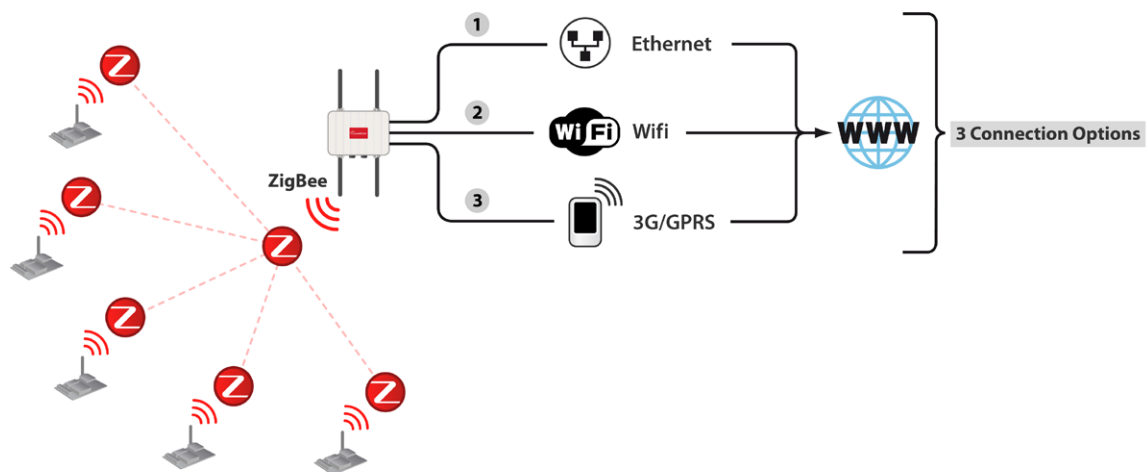
If you change any of the parameters in Waspote or Meshlium you will have to do it in both platforms so that they still can communicate.

We can perform three different storage options with the ZigBee frames captured:

- Local File System
- Local Data Base
- External Data Base



You can also send the information received to the Internet using the Ethernet, Wifi and 3G/GPRS interfaces.



Local File System

Steps:

1. Give a name to create a new file where the ZigBee data will be saved.
2. Select this file and press the "Select file" button.
3. Set the check box "Store frames in the selected file" and press the "Save" button.

From now Meshlium will automatically capture the ZigBee frames and will store the results in this file. This process will also continue after restarting Meshlium.

The file will be created in the folder `"/mnt/user/zigbee_data"` and can be downloaded just selecting it and pressing the "Download" button.

At any time you can see the last “x” lines added to the file. Just set how many lines you want to see and press the “Show data” button.

Captured Data

Local file | Local DataBase | External Database | Show me NOW

Capturer Available

File list

- default

☒ Store frames in a file Save

Select file | Create new file

Download file Delete file Show data Last lines.

```

2012-05-03 11:40:47 -mac:0013a200407791fc -x:-31 -y:51 -z:1056 -temp:29 -bat: 98%
-----
2012-05-03 11:40:49 -mac:0013a200407791fc -x:-31 -y:52 -z:1056 -temp:29 -bat: 98%
-----
2012-05-03 11:40:50 -mac:0013a200407791fc -x:-32 -y:51 -z:1057 -temp:29 -bat: 98%
-----
2012-05-03 11:40:51 -mac:0013a200407791fc -x:-32 -y:51 -z:1057 -temp:29 -bat: 99%
-----
2012-05-03 11:40:55 -mac:0013a200407791fc -x:-30 -y:52 -z:1056 -temp:29 -bat: 98%
-----
2012-05-03 11:41:01 -mac:0013a200407791fc -x:-31 -y:51 -z:1056 -temp:29 -bat: 98%
-----
2012-05-03 11:41:03 -mac:0013a200407791fc -x:-31 -y:52 -z:1057 -temp:29 -bat: 98%
```

Local Data Base

Meshlium has a MySQL data base up and running which is used to store locally the information captured. In the "Local Data Base" tab you can see the connection parameters.


- **Database:** MeshliumDB
- **Table:** zigbeeData
- **IP:** localhost / 10.10.10.1 *
- **Port:** 3306
- **User:** root
- **Password:** libelium2007

You can change the password, see the Users Manager section.

(*) Depending on the parameters set in the Interfaces section.

Captured Data

Local file | Local DataBase | External Database | Show me NOW

 Capturer Available

Connection data

Database: MeshliumDB


Table: zigbeeData

IP: localhost

Port: 3306

User: root

Password: libelium2007

 ☒ Store frames in the local data base

Last insertions.

396	2012-05-03 11:45:00	MAC: 0013a200407791fc - X: -21 - Y: 39 - Z: 1057 - Temperature: 31 - Battery: 98
395	2012-05-03 11:44:59	MAC: 0013a200407791fc - X: -21 - Y: 41 - Z: 1057 - Temperature: 31 - Battery: 98
394	2012-05-03 11:44:58	MAC: 0013a200407791fc - X: -21 - Y: 41 - Z: 1057 - Temperature: 31 - Battery: 98
393	2012-05-03 11:44:56	MAC: 0013a200407791fc - X: -22 - Y: 40 - Z: 1057 - Temperature: 31 - Battery: 98
392	2012-05-03 11:44:55	MAC: 0013a200407791fc - X: -22 - Y: 42 - Z: 1058 - Temperature: 31 - Battery: 98
391	2012-05-03 11:44:54	MAC: 0013a200407791fc - X: -21 - Y: 42 - Z: 1057 - Temperature: 31 - Battery: 98
390	2012-05-03 11:44:42	MAC: 0013a200407791fc - X: -23 - Y: 38 - Z: 1058 - Temperature: 31 - Battery: 98
389	2012-05-03 11:44:40	MAC: 0013a200407791fc - X: -22 - Y: 39 - Z: 1057 - Temperature: 31 - Battery: 98
388	2012-05-03 11:44:39	MAC: 0013a200407791fc - X: -21 - Y: 40 - Z: 1057 - Temperature: 31 - Battery: 98

Steps:

1. Set the check box "Store frames in the local data base" and press the "Save" button.

From this time Meshlium will automatically perform Scans and will store the results in the Local Data Base. This process will also continue after restarting Meshlium.

At any time you can see the last "x" records stored. Just set how many insertions you want to see and press the "Show data" button.

External Data Base


Meshlium can also store the information captured in an External Data Base.

Steps:

1. Pressing the “Show sql script” you will get the code needed to create the data base along with the table and the right privileges.

Captured Data

Local file | Local DataBase | External Database | Show me NOW

 Capturer Available

Connection data
Database:
Table:
IP:
Port:
User:
Password:

☐ Store frames in the external data base
 Last insertions. (to create database table)

Just copy paste:

```
CREATE database MeshliumDB;
```


Just copy paste:

```
CREATE TABLE IF NOT EXISTS `zigbeeData` (
  `ID_frame` int(11) NOT NULL auto_increment,
  `TimeStamp` timestamp NOT NULL default CURRENT_TIMESTAMP,
  `mac` varchar(16) collate utf8_unicode_ci NOT NULL,
  `x` varchar(16) collate utf8_unicode_ci NOT NULL,
  `y` varchar(16) collate utf8_unicode_ci NOT NULL,
  `z` varchar(16) collate utf8_unicode_ci NOT NULL,
  `temp` varchar(16) collate utf8_unicode_ci NOT NULL,
  `bat` varchar(16) collate utf8_unicode_ci NOT NULL,
  `frame` varchar(200) collate utf8_unicode_ci NOT NULL,
  PRIMARY KEY (`ID_frame`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci AUTO_INCREMENT=39 ;
```

2. Insert this code in your MySQL management application.
3. Fill the Connection Data fields with the information about where the data base is located (IP, Port) and with the authentication options (Database, Table, User, Password).
4. Now press the “Check Connection” button to see if the configuration is correct.

Captured Data

Local file | Local DataBase | External Database | Show me NOW

 Capturer Available

Connection data

Database:

Table:

IP:

Port:

User:

Password:

☐ Store frames in the external data base

Last insertions. (to create database table)


Connecting to the database server ...
Selecting database ...
Sending Inquiry ...
Query generated with id: 39
OK

- Set the check box "Store frames in the selected file" and press the "Save" button.

From this time Meshlium will automatically perform Scans and will store the results in the Local Data Base. This process will also continue after restarting Meshlium.

At any time you can see the last “x” records stored. Just set how many insertions you want to see and press the “Show data” button.

Captured Data

 Capturer Available

Local file
Local DataBase
External Database
Show me NOW

Connection data

Database:


Table:

IP:

Port:

User:

Password:


☒ Store frames in the external data base

Last insertions.

 (to create database table)

360	2012-05-03 11:43:16	MAC: 0013a200407791fc - X: -17 - Y: 37 - Z: 1059 - Temperature: 30 - Battery: 98%
359	2012-05-03 11:43:15	MAC: 0013a200407791fc - X: -18 - Y: 37 - Z: 1057 - Temperature: 30 - Battery: 98%
358	2012-05-03 11:43:11	MAC: 0013a200407791fc - X: -17 - Y: 37 - Z: 1057 - Temperature: 30 - Battery: 98%
357	2012-05-03 11:43:10	MAC: 0013a200407791fc - X: -18 - Y: 36 - Z: 1058 - Temperature: 30 - Battery: 98%
356	2012-05-03 11:43:08	MAC: 0013a200407791fc - X: -18 - Y: 36 - Z: 1058 - Temperature: 30 - Battery: 98%
355	2012-05-03 11:43:03	MAC: 0013a200407791fc - X: -19 - Y: 37 - Z: 1057 - Temperature: 30 - Battery: 98%
354	2012-05-03 11:42:51	MAC: 0013a200407791fc - X: -19 - Y: 36 - Z: 1057 - Temperature: 30 - Battery: 98%
353	2012-05-03 11:42:44	MAC: 0013a200407791fc - X: -18 - Y: 35 - Z: 1057 - Temperature: 30 - Battery: 98%
352	2012-05-03 11:42:43	MAC: 0013a200407791fc - X: -19 - Y: 36 - Z: 1058 - Temperature: 30 - Battery: 98%

Show me now!

In the "Show me now!" tab you can see in real time the Scans captured.

You can specify if you want the information to be updated periodically with the defined interval just checking the "Use the Defined Interval" button.

Captured Data

Local file Local DataBase External Database **Show me NOW**

Scan interval

Thu May 3 13:03:02 GMT 2012

MAC: 0013a200407791fc
Acc-X: -47
Acc-Y: 69
Acc-Z: 1055
Temperature: 29
Battery: 98%

- **Capturing and storing your own ZigBee frames**

Their own ZigBee frames are stored in the local file and the databases with the prefix Frame, but you can create your own capture and storage applications.

We have created two template code files in order to help developers to create their own capture and storage applications.

The default ZigBee “capturer and storer” comes with the next structure (pseudocode):

```
main()
{
    init();                //initializes the serial port to communicate with the XBee radio
    while(1)
    {
        getFrame();        //read the frame received from the ZigBee radio
        parseFrame();      //parse the frame in order to extract the data
        storeInAFile(fileName); //store the data in a file (if enabled)
        storeInADB(LOCAL);  //store the data in the local DB (if enabled)
        storeInADB(EXTERNAL); //store the data in an external DB (if enabled)
    }
}
```

The real files (zigbeeStorer.h, zigbeeStorer.c) can be downloaded from the Meshlium Development section:

<http://www.libelium.com/development/meshlium>

You can download these files and change them in order to make it compatible for your specific sensor configuration.

Compilation:

The compilation can be done in the same Meshlium. Just copy these files in a folder accessing by SSH and execute:

```
$ gcc -o zigbeeStorer -I/usr/include/mysql zigbeeStorer.c -L/usr/lib/mysql -lmysqlclient -Wall
```

Important: before starting to execute the new binary “zigbeeStorer” stop the default ZigBee daemon:

```
$ /etc/init.d/ZigbeeScanD.sh stop
```

If you want your own ZigBee capturer application to execute each time Meshlium starts you have to change the daemon file (/etc/init.d/ZigbeeScanD.sh) in order to execute your binary.

You will find support in the Libelium Forum at: <http://www.libelium.com/forum>

• Sending ZigBee frames from Meshlium to Wasmote

Meshlium can also send ZigBee frames to the Wasmote nodes. In order to use this feature you have to stop the “capturing and storing” daemon which is running in the system.

To do so access by SSH to Meshlium and stop the default ZigBee daemon::

```
$ /etc/init.d/ZigbeeScanD.sh stop
```

Now you can execute the ZigBeeSend command. There are several ways to send information to a node:

- Using its 802.15.4 MAC address (64b)
- Using its Network address (MY) (16b)
- Performing a broadcast transmission

Sending to Wasmote using its MAC address (64b):

```
$ ./ZigBeeSend -mac 0013a2004069165d "Hello Wasmote!"
```

Sending to Wasmote using its Net address (MY - 16b):

```
$ ./ZigBeeSend -net 1234 "hello Wasmote!"
```

Send to all the Wasmote devices at the same time - Broadcast mode:

```
$ ./ZigBeeSend -b "hello everybody!"
```

The source code “ZigbeeSend.c” and the reception program to be installed in Wasmote can be downloaded from the Meshlium Development section: <http://www.libelium.com/development/meshlium>

You can download these files and change them in order to get new features and sending options.

Compilation:

The compilation can be done in the same Meshlium. Just copy these files in a folder accessing by SSH and execute:

```
$ gcc -o ZigBeeSend ZigBeeSend.c -lpthread
```

Important: If you want to create a “ZigBee sending” daemon that is executed each time Meshlium starts you have to deactivate the “ZigBee Capturer” daemon (/etc/init.d/ZigbeeScanD.sh) as the ZigBee radio has to be used by one process at a time.

You will find support in the Libelium Forum at: <http://www.libelium.com/forum>