Towards the Internet of Things: A Real-World Testbed

Nacer Khalil

School of Science and Engineering

Al Akhawayn University in Ifrane

Email: n.khalil@aui.ma

Mohamed Riduan Abid

School of Science and Engineering

Al Akhawayn University in Ifrane

Email: r.abid@aui.ma

Driss Benhaddou

School of Engineering and Technology

University of Houston

Email: dbenhaddou@uh.edu

**Abstract**

***The Internet is smoothly migrating from an Internet of people to an Internet of Things. By 2050, it is expected to have 50 billions “Things” connected to the Internet.***

***However, such a migration induces a strong level of complexity when handling interoperability between the heterogeneous things, e.g., Wireless sensor networks, RFID, and mobile computing. In this context, a couple of standards have been already set, and they are currently reached the mature stage, e.g. IPv6, 6loWPAN, M2M, and IMS.******In this paper, we focus on the integration of wireless sensor networks to the Internet. We shed further on the subtleties of such integration, and we deployed a real-world test-bed where wireless sensors are used to control electrical appliances in a smart building.***

**1. Introduction**

The internet of things is the future of internet. It is an internet where physical devices can communicate by producing and consuming information. This make these devices more intelligent as they will be able to provide services but at the same time consume others. These physical devices will be uniquely identified in the internet by making use of technologies such as IPV6 and RFID. There will be millions of devices in the internet both producing and consuming information [8]. To build the internet of things, there are numerous things that should be added to the actual internet to make all this possible. The first thing is to add the TCP/IP protocol stack to the physical devices and wireless sensor networks. Making use of the 6lowpan technology does this. The identification of each physical object is possible by using technologies such as RFID which is a low power identification chip that can allow you to identify each object uniquely. Concerning communication between physical objects, M2M technology provides standards to make the communication possible and robust.

To make these physical devices cyber physical system they will not only need to communicate but also to sense the environment and this is possible when making use of the WSN that allow physical devices to sense the environment based on their requirements.

In this paper, we create a small network where a machine-to-machine communication is possible. This can be part of what is known as Internet of things [1]. The deployed tested is composed of a wireless sensor network(WSN), a middleware station and a mobile client within the field of smart grids where data is collected from the motes within the WSN and sent to the middleware. The mobile client accesses this data and the mobile client can send messages directly to any mote within the WSN network in order to turn on or off any appliance. The contributions of this paper are:

• The system is developed using heterogeneous networks, different data link layer technologies

• The WSN uses IPV6 over 6lowPAN whereas other components of the system use IPV4

To solve these issues, different programs have been used to deal with these issues.

The Rest of the paper is organized as follows…. Section 2 presents related work. In Section 3, we provide a review of the internet of things concepts and issues. Section 4 presents 6lowpan technology for low power devices such as wireless sensor networks that is used in this paper. Section 5 presents the system architecture of the testbed. In section 6 the implementation of the system is discussed more into details and section 7 presents the the measurements that are done on the testbed.

## 2 Literature review

Smart grid as being one of the most important applications of the internet of things makes use of different technologies such as Zigbee in the WSN [2]. [3] discusses the need to introduce IPV6 within the wireless sensor networks and discusses the the existing approaches as well as the issues related to introducing IPV6 on top of Zigbee. Such issues are fragmentation, frame size, addressing, and security issues. This paper introduces a real testbed that includes the whole TCP/IP protocol provided by the BLIP project that takes into consideration most of those issues and implement the two-way communication as needed by smart grids and measures the performance of such a system. El Kouche [4] also investigates on the widely used architectures and technologies and explains which architecture is the most suitable for a WSN within the internet of things [4]. [5] discusses the requirement of the IOT gateway as well as a proposed architecture for the system to be deployed in the gateway. A rather similar architecture to what is done in this paper is proposed by Yerra, Bharathi, Rajalakshmi and Desai [6]. They also use GSM technology to transmit information whereas in our case, internet is used for this purpose.

## 3 Internet of things

The internet of things is a new concept based on the idea that there will be much more users that are computers, machine than human beings. This means that machines will play an important role in the internet and they will be able to communicate autonomously without the interaction of human beings. Wireless sensor network will be part of the internet of things and therefore will be actively communicating in the internet by sending sensor data. Nowadays, the internet is mainly based on a human-to-human. In the future of internet called the internet of things, we will have in addition to human-to-human communication, human-to-machine communication and machine-to-machine communication. An example of human-to-machine communication is when the human will be able to control the wireless sensor network’s sampling time. An example of machine-to-machine communication which will be discussed in this paper is the middleware being able to turn off the light because no one is in the room. The middleware by making use of the sensor data will communicate with motes to turn on or off the light bulb.

Ubiquitous computing and pervasive computing is an important component of the Internet of things. Physical objects will be able to communicate in the Internet by producing information as well as consuming information [9]. The main technological components needed in the Internet of things are the Radio Frequency identification (RFID) and is a technology that enables physical devices to identify themselves through this low power technology. Another important technological component is the wireless sensor network (WSN) and it is a technology that gives the physical objects to sense the environment and communicate such information. Mobile computing and cloud computing are important components as most of the storage and processing power will come from the cloud that will provide such services to the different physical devices in the internet of things.

The internet of things has a great impact on the every day life of people as it will affect our lives inside our homes where all physical devices will be able to communicate information by providing services and consuming others. More tasks will be automated, as most of the physical devices will communicate their needs to each other. Smart grid is another important technology made possible with the Internet of Things. The US national intelligence listed the Internet of things as being on six disruptive civil technologies with potential impact on the US national power [10].

## 4 Linking the Things: 6loWPAN

6loWPAN stands for "IPv6 over Low power Wireless Personal Area Networks". It stands for the name of the working group within IETF. It is based on the idea that all the devices independently of their processing power or energy resources should be able to have the TCP/IP protocol stack and be part of the internet of things. In order to be able to build the TCP/IP protocol stack in low-power devices, there have been multiple aspects of the protocol stack that has to be optimized. As an example, the IP MTU(maximum transmission unit) is fixed at 1280 bytes whereas the Zigbee MTU is only 127 bytes. This means that not every IPV6 packet can be inserted within a Zigbee frame. Another issue is related to the addressing with the 128 bits address. in 6lowpan, the addressing via the IPV6 address is performed hierarchically to first identify is the packet is sent to a device within that network but afterwards, addressing is done by counting mainly on the 16 last bits of the IPV6 address. These were just two of a large set of issues that 6lowpan tries to solve in order to enable even the low-power devices to join the internet of things. TinyOS as being one of the most used operating system comes with its implementation of 6lowpan through a project called BLIP (Berkeley Low-power IP stack) that provides a lightweight implementation of the 6lowpan and that can be installed and used with a variety of motes as long as they support IPV6. This project makes use of BLIP to provide the TCP/IP protocol stack to the devices within the wireless sensor network.

## 5 System architecture

The Testbed is composed of three building blocks.

• Wireless sensor network (WSN)

• Gateway server

• Middleware

• Mobile client

The WSN communicates using Zigbee as well as with the gateway server using IPV6 technology whereas the communication between the gateway server, the middleware and the mobile client is done via Wi-Fi using IPV4 technology. The architecture enables any component within this system to communicate with any other system independently of the communication medium used (Zigbee or Wi-Fi) or the network technology used (IPV4 or IPV6). Figure 1 is a network diagram that depicts the different components of the system as well as the routes that exist for any component to communicate with another.

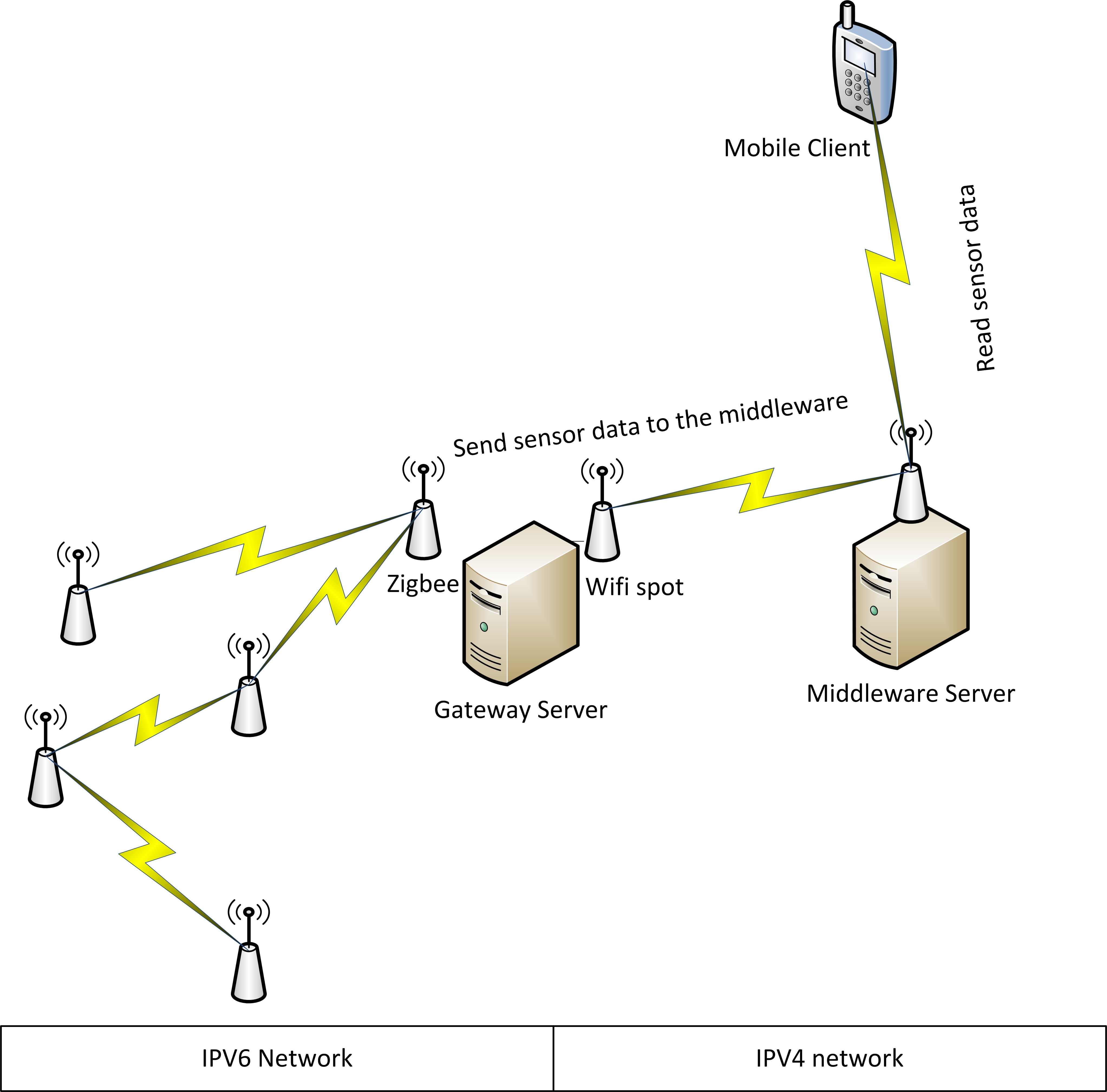


Figure 1: Network diagram

### 5.1 Wireless sensor network

The hardware is composed of motes of type crossbow MPR2600 that are used to build the wireless sensor network as shown in figure 2.From the network aspect, the wireless sensor network is based on the mesh topology. It is also designed in order to be an ad-hoc network. This means that you can place the mote any where as long as there is one link it can attain to communicate with. The routing is multi-hop and links are created dynamically between different motes of the WSN provided that there frames can reach the destination.



Figure 2: Crossbow MRP2600

### 5.2 Gateway server

The gateway server is a crucial component within the system. It is responsible for extracting Wi-Fi frames and forwarding them as Zigbee frames and vice versa. It is also responsible for receiving IPV4 packet and transforming them into IPV6 and vice versa. The gateway server has other functionalities which are the ability to receive sensor data from all the motes in the wireless sensor network and forwarding them to the middleware. In case the link between the gateway server and the middleware is lost, the gateway server stores the received data and sends them when the link is up again. This feature serves as a way to avoid losing data.

### 5.3 Middleware

The middleware is a special program that is used to hide the heterogeneity of the system from external users. The middleware also provides automation mechanisms to control and reduce the energy consumption, but the main features are the ability to receive data, filter it, transform it and store it in a coherent fashion in order to make the most of this data and use it smartly to reduce consumption. Concerning the client access, the middleware provides a set of web services that enable the clients to access all sorts of information such as the real-time consumption, daily consumption and other information inferred from the sensor data coming from the wireless sensor network.

### 5.4 Mobile client

The mobile client application is an interactive application for android phones that enable users to get access to the real-time consumption of their homes and also to remotely control their appliance by turning on and off any appliance linked to some mote in the wireless sensor network. The mobile client when wanting to turn on or off some appliance sends a request directly towards the mote responsible for that action and designate the mote using its "virtual" IPV4 address that it does not own in really since it has an IPV6 address but that virtual IPV4 address is reserved for that mote and the translation is dealt with at the level of the gateway.

Now that all components have been explained, the data flow of the information is to be explained. As it was stated before, the any component of the system can communicate with any other independently of the data link layer technology or network layer technology.

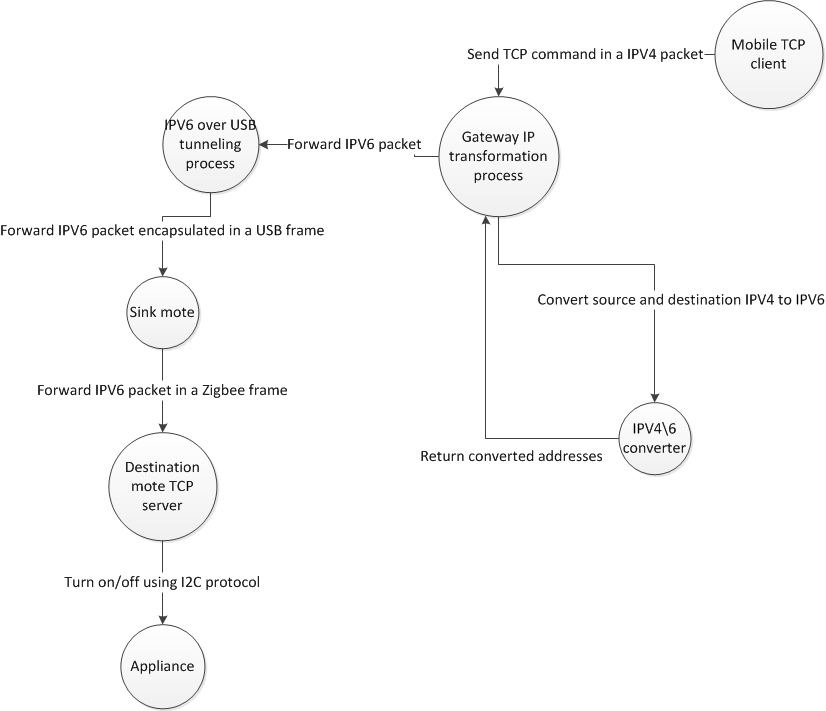


Figure 3: Data flow diagram for the mobile client sending on/off commands (Re-adjust the picture, size and clearness)

As one of the main goals of this project is to provide a two-way communication between the client and sensor, we will explain both way’s flow of information. Figure 3 depicts the flow of information going from the client to the appliance. As the mobile client in the system is connected to a Wi-Fi network that uses IPV4 and the mote is a network that uses a IPV6, then there should be a process that does some transformation to incoming and outgoing packets. So the client sends the packet to the virtual IPV4 address of the mote. The gateway receives it translates the virtual IPV4 address into the real IPV6 address of the mote and sets as source address the virtual IPV6 address of the mobile client. The new IPV6 packet is created holding the same payload as the original IPV4 packet. This new IPV6 packet is to be forwarded to the wireless sensor network. To do so, it is passed to the tunnel that encapsulates the packet into a USB frame and sends it to the mote sink that extracts the IPV6 packet from the USB frame and encapsulates it into a Zigbee frame. Once the Zigbee frame arrives to the destination mote, the TCP datagram is extracted from the IPV6 packet and passed to the TCP server of the mote that reads the message and executes it by turning on/off the appliance using I2C protocol.

As a second part of the two-way communication, the mote sends periodically sensor data to the middleware but again the communication is goes through several steps as depicted in figure 4. The mote reads periodically sends sensor data from the sensor, transforms the data and sends it. To send it the mote client connects to a TCP server hosted at the gateway. The IPV6 packet is encapsulated in a Zigbee frame that is forwarded to the mote sink that extracts the IPV6 packet and encapsulates it into a USB frame and forwarded to the gateway that reads the TCP datagram. Once the sensor data is at the gateway, the gateway forwards the sensor data to the middleware and if the link is down, the sensor data is temporarily stored in a database hosted in the gateway and once the link is up, all the stored data is sent to the middleware and cleared from the database.

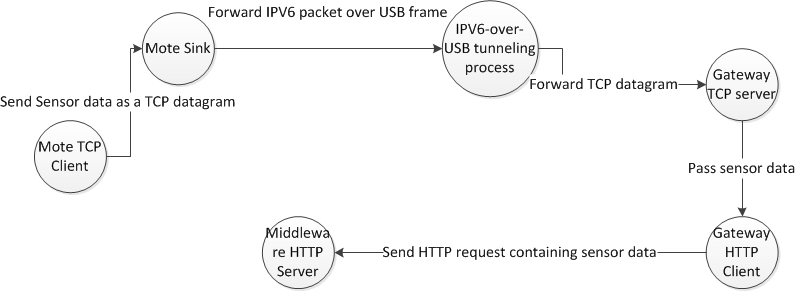


Figure 4: Data flow diagram for the mobile client sending on/off commands (same here as well)

## 6 Implementation

To have a system that is capable of providing a two-way communication between any host and the any mote within the wireless sensor network, four building blocks had to be provided and which are the following:

• Mote programming

• Mote sink packet forwarding

• Network gateway packet transformation

• Network gateway sensor data server

### 6.1 Mote programming

Each mote within the WSN network is equipped with a current transformer that is attached to data acquisition board through which data is read and transformed to the appropriate format and sent to the network host. In addition to this, the appliance is attached to the mote through the relay pins existing within the data acquisition board in the mote. In other words, the mote can control the electricity going to each mote and can allow it or block it. This means that one can control the appliance by using some of the functionalities provided by the mote. From the mote’s perspective there are two parts that are implemented within its TinyOS program. A TCP server that is used to receive on/off requests in order to control the mote and a TCP client used to send sensor data.Once the program installed, an IPV6 address is passed to the installation routine in order to assign a static IPV6 address to the mote in which the program is cross-compiled and installed. The TCP server and TCP client work in parallel as each one’s traffic is handled separately.

#### 6.1.1 TCP server

The TCP server is an important component in the mote’s program. Any one willing to control the appliance must connect to the TCP server and send requests. As a mote may control more than one appliance, we then identify the appliance by a unique id and send a zero to turn off or one to turn on. Requests such as "1 1" means to turn on the appliance with id 1.

#### 6.1.2 TCP Client

The TCP Client is an important component in the mote’s program. It serves as mean to send sensor data to the gateway reliably. Once the mote is turned on, the client connects to a TCP server that is in the gateway, then the consumption data is sensed periodically (once per second) and sent to the TCP server who deals with the sensor data.

### 6.2 Mote sink packet forwarding

The mote sink packet forwarding module is a special program installed within a mote that is equipped with a USB port that plays the role of a network interface card. The mote is attached to the gateway station and has the BLIP (Berkeley Low power IP stack Protocol) module within in. In addition it communicates with the gateway using USB protocol. In the gateway station, the network interface module is a sort of a IPV6 over USB tunnel which means that if an IPV6 packet packet is going to the sink, it is encapsulated within a USB frame and once it arrives to the mote, the IPV6 packet is extracted and forwarded to the destination mote holding that IPV6 address. The other way around is fairly similar, when a mote wants to send an IPV6 packet to the outside world, the mote creates the packet, sends it to the sink that forwards it by encapsulating it into a USB frame.

### 6.3 Network gateway packet transformation

This is a very important component in the system. The issue is the following. The WSN network supports only IPV6 while other components such as the middleware and the mobile client do not necessary have an IPV6 address, but we still want all the components to communicate independently of the IP technology to be used. To do so, we have created a network packet transformation program. This program basically converts IPV4 to IPV6 and vice versa. To do so, it assigns virtual IPV4 addresses to IPV6 address holders and and IPV6 address to IPV4 address holders. With such a program, each player in the network has both an IPV4 and an IPV6 but is aware of only the one that is assigned to him. The other "virtual" address is known only at the level of the program installed at the gateway station between the WSN and the outside world.

The flow of information works as follow; when a station wants to send requests to a mote, it will send IPV4 packet holding the request to the mote but by addressing it using its virtual IPV4 address. This packet will be sniffed by this packet transformation program that will extract the TCP datagram from the ip packet, create a new IPV6 packet specifying the source address as the virtual address of the host and the destination as the address of the mote and then append the TCP datagram to the newly created packet and send it to the mote sink packet forwarding component that is seen by the gateway as a network interface card. This leads to a complicated issue that is to be handled and which is that the program should keep track of the request responses in order to forward them correctly to the destination.

To solve such a problem, an algorithm has been created whose sole role is to mechanically compute the IPV6 address of the host based on its IPV4 and vice versa. This algorithm is based on function whose primary quality is the fact that it is bijective which means any IPV4 address is mapped to one and only one IPV6 function and vice versa. So whenever a request is coming, the source and destination address will be converted using this algorithm and this will make avoid the whole request response tracking part.

## 7 Measurements

The goal of this part is to measure to what extent is the system able to operate reliable and offering some acceptable level of performance. As this system is to be deployed in a small environment such as an appartment or a house where each mote will be connected to one or more appliances, and motes will be closely positionned and this positioning will make it possible to make use of the multihop routing and therefore placing no more than one sink in each house. In this expirement, we try to simulate the normal work of this system where each sensor is read once every second generating traffic in the zigbee network. Our goal is to measure the delay and see how does it vary in this system with the amount of traffic generated. As the multihop routing protocol gives more priority to packets to be routed than those addressed to that mote, we would like to see if this prioritization reduces the delay when having more and more motes sending packets through the network. Table 1 illustrates how the delay and jitter varies as we add more and more nearby motes to the network.

|  |  |  |
| --- | --- | --- |
| # of motes | Average Delay (ms) | Jitter (ms) |
| 1 | 87 | 1.2 |
| 2 | 88 | 2.0 |
| 3 | 88 | 2.7 |
| 4 | 88 | 3.1 |
| 5 | 90 | 3.3 |
| 6 | 89 | 3.5 |

It is clear that the average delay is not much affected by the number of motes in the network, but the jitter on the other hand is clearly affected by the number of motes in the network. This is mainly due to the fact that there are timeframes where the network is less busy and other timeframes where the network becomes busier. This is why the jitter grows as the number of motes in the network grows with it.

## 8 Conclusion

The Internet of things is the next revolution in computing. The testbed presented in this paper illustrates how can smart grid by making use of numerous technologies such as TinyOs based WSN, 6lowPan can integrate within the Internet of things and benefit from it. The test bed presents the 2-way communication as needed by smart grids and also discusses the architecture of the system and the issues that were handled to make the communication possible and smooth.

The issues that were handled are mainly related to making the communication possible even if it is on a homogeneous system where the data link layer is not the same and the network layer is not the same. The packet is transmitted from the an IPV4 network based on the Wifi network and moves to an IPV6 network based on USB then to Zigbee. These were one of the numerous issues handled in this paper.

Smart grids can and will benefit from the internet of things and making it as one of the most important applications of the internet of things and therefore enable more efficiency in the generation and consumption of data.

**References**

[1] Q. Ou, Y. Zhen, X. Li, Y. Zhang, and L. Zeng, “Application of internet of things in smart grid power transmission,” in *Mobile, Ubiquitous, and Intelligent Computing (MUSIC), 2012 Third FTRA International Conference on*, 2012, pp. 96–100.

[2] L. Coetzee and J. Eksteen, “The internet of things - promise for the future? an introduction,” in *IST-Africa Conference Proceedings, 2011*, 2011, pp. 1–9.

[3] C.-W. Lu, S.-C. Li, and Q. Wu, “Interconnecting zigbee and 6lowpan wireless sensor networks for smart grid applications,” in *Sensing Technology (ICST), 2011 Fifth International Conference on*, 2011, pp. 267–272.

[4] A. Kouche, “Towards a wireless sensor network platform for the internet of things: Sprouts wsn platform,” in *Communications (ICC), 2012 IEEE International Conference on*, 2012, pp. 632–636.

[5] Q. Zhu, R. Wang, Q. Chen, Y. Liu, and W. Qin, “Iot gateway: Bridgingwireless sensor networks into internet of things,” in *Embedded and Ubiquitous Computing (EUC), 2010 IEEE/IFIP 8th International Conference on*, 2010, pp. 347–352.

[6] R. Yerra, A. Bharathi, P. Rajalakshmi, and U. Desai, “Wsn based power monitoring in smart grids,” in *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2011 Seventh International Conference on*, 2011, pp. 401–406.

[7] =2plus 43minus 4J. Martìn, I. Gómez, E. Montoya, J. Song, J. Olloqui, and R. Martìnez, “Batnet: an implementation of a 6lowpan sensor and actuator network,” in *Proceedings of the fourth international conference on Future energy systems*, ser. e-Energy ’13.1em plus 0.5em minus 0.4emNew York, NY, USA: ACM, 2013, pp. 281–282. [Online].

[8] M. Yun and B. Yuxin, “Research on the architecture and key technology of internet of things (iot) applied on smart grid,” in *Advances in Energy Engineering (ICAEE), 2010 International Conference on*, 2010, pp. 69–72.

[9]  M. Jung, C. Reinisch, and W. Kastner, “Integrating building automation systems and ipv6 in the internet of things,” in *Innovative Mobile and In- ternet Services in Ubiquitous Computing (IMIS), 2012 Sixth International Conference on*, 2012, pp. 683–688.