Secure Sensor Node with Raspberry Pi

Soham Banerjee[#], Divyashikha Sethia[*], Tanuj Mittal[*], Ujjwal Arora[*], Akash Chauhan[*]

[#]*Department of Electrical Engineering,* [*]*Department of Computer Engineering, Delhi Technological University*
Main Bawana Road, Delhi, India

[1]soham.dce@gmail.com, [2]divyashikha@dce.edu, [3]tanuj.183@gmail.com, [4]ujjwalarora1992@gmail.com,
[5]akash75457@gmail.com

***Abstract*—Information Technology, when implemented in healthcare, has the potential to radically improve the quality and efficiency of services being made available to patients. With the advent of mobile computing and wireless sensor networks, data acquisition and processing has become much faster and less expensive. Since there is an increase in possible software attacks on privacy and safety of health applications, safety and reliability of the sensor data is an important issue that needs to be addressed in this field. In this paper, a unique design of a secure sensor node prototype has been proposed and implemented, which communicates over Bluetooth using RC4 encryption algorithm between a mobile phone and their monitoring equipment. The design uses an accelerometer based sensor, which can be used as a prototype for a body sensor for fall detection in elderly people [1]. The data from the sensor is processed, encrypted and wirelessly communicated by Raspberry Pi (a Single Board Computer) to a mobile phone. The security issues have been addressed in two ways. Firstly, the loss of data is prevented by introducing a wired connection between the Raspberry Pi and the accelerometer based sensor. Secondly, a secure data communication is ensured by encrypting the sensor data using an encryption key sent from the mobile phone using Bluetooth. Our prototype, can be used to investigate and design several new sensor nodes with different interfaces, encryption and compression algorithms. The future work extends the application of this secure sensor node to the design of the body sensor module.**

Keywords- Secure Sensor Node, Raspberry Pi, Single Board Computer, Accelerometer, I2C, Bluez, Cryptography, RC4, Smartphone

I. INTRODUCTION

According to a study conducted by The Telegraph, UK [2] within the next decade, for the first time in human history there will be more people aged 65 years and older, than children under five in the world. A notable percentage of this population will reside in nursing homes and hospitals in times to come. In the future, these residences will employ pervasive networks which will provide continuous medical monitoring, control of home appliances, medical data access, and emergency communication. The body sensor module will be extremely beneficial to the patients and their caregivers who will be able to monitor their health on their own. It will enable doctors and physicians to remotely monitor their patients' health and improve the quality of healthcare, increasing safety and reducing the overall cost of healthcare incurred by the patients.

Body sensor networks monitor health parameters using body sensor devices [3]. These devices are useful in biometric and medical applications for real-time monitoring of a patient's state or for acquiring sensitive data which can be subsequently analyzed to provide a medical diagnosis. The initial motivation of the work was to develop a secure sensor node prototype for developing secure body sensors. Most of the sensors in the market overlook the security aspect. Rasberry Pi has been used for initial deployment of a secure body sensor based on accelerometer which can be used as a fall detection sensor in elderly people. We have used accelerometer sensor for initial development of a prototype since we can get the changes in the readings promptly. With this design the sensors for temperature, bold pressure, oxymeter etc can similarly be incorporated in the design to gather vital health parameters.

Raspberry Pi has been chosen rightly as the single board computer for this application because it has the highest performance to cost ratio and is one of the smallest single board computers available in the market. This paper describes in detail, the components, design and functioning of one secure sensor node prototype which has been developed, tested and verified for the development of a body sensor module. The sensor information gathered can be communicated wirelessly to the mobile phone using any of the following three connectivity options: Bluetooth, Bluetooth light or NFC, depending on the application. In this project, Bluetooth technology has been used.

The paper is broadly divided into five sections. Section II provides an overview of the various components used in the design of the secure sensor node prototype. Section III describes the implementation of the secure sensor node prototype in detail which includes connections, sensor data acquisition, the RC4 encryption algorithm and its comparison to other encryption algorithms, communication of the secure sensor node prototype with the mobile phone over Bluetooth and some screen snapshots are presented. Section IV gives a brief description about some related work already done. We conclude in Section V, which gives the conclusion and the future scope that can be incorporated in this project.

978-1-4799-1205-6/13/$31.00 ©2013 IEEE

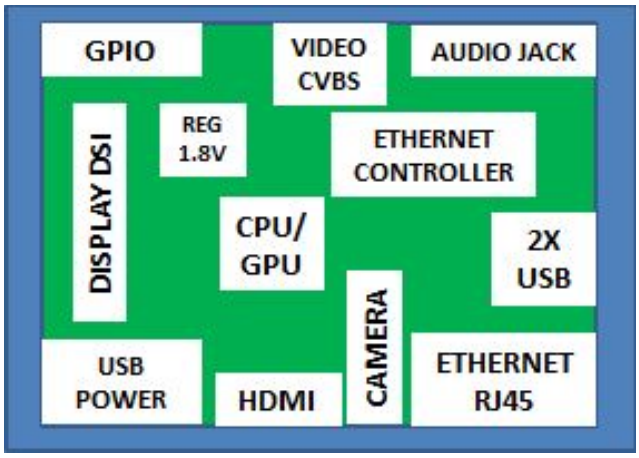

Figure 1. Internals of Single Board Computer(SBC)

TABLE I. COMPARISON BETWEEN MOST SINGLE BOARD COMPUTERS

Features	BeagleBone	Raspberry Pi	Panda Board
Single-core/Multi-core frequency	Single-core/ 720 MHz	Low power single-core/ 700 MHz	Dual-core/ 1 GHz
Area	45 cm^2	45 cm^2	115 cm^2
RAM	256 MB	512 MB	1 GB
Price	$133	$35	$500

II. COMPONENTS OF THE SECURE SENSOR NODE

A unique design and implementation of a secure sensor node has been carried out based on three major components: a single board computer, an accelerometer based sensor-ADXL345 and a Bluetooth Dongle. Subsidiary components have been used to setup the secure sensor node prototype.

A. Major Components

1) Single Board Computer

The choice of the single board computer depends on the kind of application which the sensor node uses. Here, we are using it to design a body sensor module which will interface a number of sensors. Raspberry Pi has been chosen for this application [9]. Table 1 gives the comparison between the two most used single board computers in market to justify the use of Raspberry Pi.

Among other SBCs, Raspberry Pi is the cheapest single board computer available with the best performance/cost and RAM/cost ratio. Its small size, low cost, low power consumption and high processing power makes it suitable for the design of this body sensor.

2) Accelerometer, ADXL345

The ADXL345 [4] is a 3-axis accelerometer with a high resolution (13-bit) measurement. It covers a range of ±16 g. The output data present in the data register is formatted as 16-bit 2's complement and can be accessed through an SPI (4- or 3-wire) or a I2C digital interface. The ADXL345 is small thin and low-power, hence it is suitable for mobile device applications. It measures static acceleration due to gravity in tilt-sensing applications, and also dynamic acceleration resulting from motion or shock. It has a high resolution (3.9 mg/LSB) which enables measurement of inclination changes of less than 1.0°.

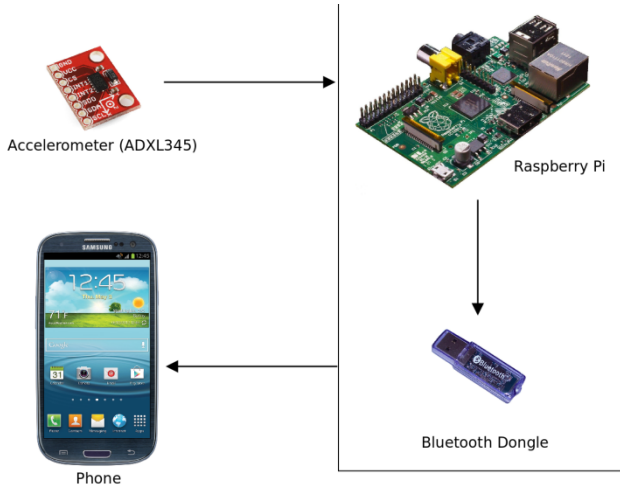

Figure 2. Flow diagram of body sensor

It comes with a 32-level first in, first out (FIFO) buffer which can be used to store data to lower overall system power consumption[4]. This sensor module is chosen since it provides high accuracy and less complex sensor data. Bluetooth Dongle

An external Bluetooth module in the form of a USB dongle is used. This Bluetooth dongle manufactured by DELL has a range of 100m and data rate up to 3 Mbps and supports USB 2.0.

B. Subsidiary Components

A powered USB hub consisting of four USB ports, a micro USB port (to connect to the USB port of the Pi) and a power jack (to provide a connection to an external power source) is used. It is used during the design of the secure sensor node prototype to connect keyboard and mouse.

A monitor, an HDMI to VGA converter, a power source(Samsung Charger)with output current rating of 700mA-1000mA and output voltage rating of 5V, a USB keyboard, a USB mouse, a PCB board and 8 single pin connectors are other subsidiary components that are used during the design of the secure sensor node prototype.

III. IMPLEMENTATION OF THE SECURE SENSOR PROTOTYPE

Fig. 2 describes the flow of data within the secure sensor node prototype and also provides a lucid illustration of the connections in it.

A. Connections of the Secure Sensor Node Prototype

The accelerometer based sensor consists of eight pins, two of which are the power and ground pins. Two Interrupt pins

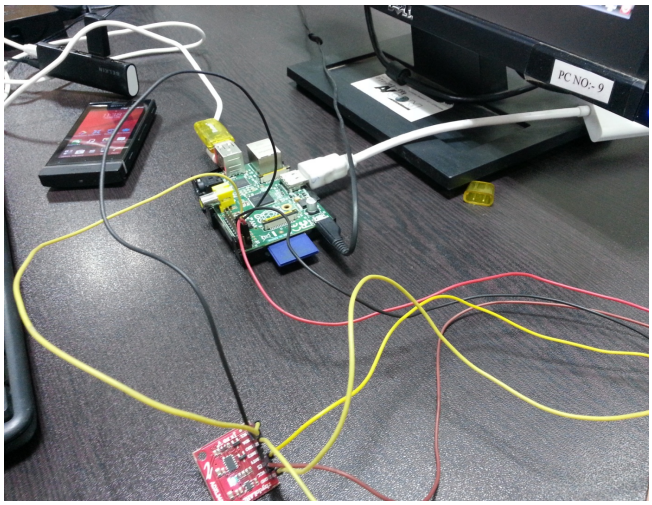

Figure 4. Real image of connections of body sensor

are available but are left unconnected. The CS' and VDD pins are supplied with 3.3 V from the Raspberry Pi. The SDO and the GND pin of the accelerometer based sensor are connected to the GND pin of the Raspberry Pi. The SDA pin for data interchange is connected to the third pin of the Raspberry Pi and the SCLK pin to the fifth pin of the Raspberry Pi. Communication between the accelerometer based sensor and the Raspberry Pi can take place using either of the two serial protocols: SPI [4] or I2C [4].

The accelerometer based sensor (ADXL345) supports standard (100 kHz) and fast (400 kHz) data transfer modes if the given timing parameters are met.

I2C address of 0x53 (followed by the R/W bit) is chosen by grounding the SDO/ALT ADDRESS pin. If other devices are connected to the same I2C bus, these devices cannot have an operating voltage exceeding VDD I/O by more than 0.3 V [4]. Therefore external pull-up resistors, Rp are necessary for proper I2C operation. The I2C driver [5] is enabled and the I2C modules [5] and the python-smbus module [5] are downloaded and installed. Fig. 3 shows how the Raspberry Pi is connected to the accelerometer based sensor in this project. The Program Flow of the secure sensor prototype is shown in Fig. 4.

B. Receiving Data from the Accelerometer Based Sensor

The accelerometer based sensor is set to measurement mode. The range of measurement of the accelerometer based sensor is set according to the user. After this, the 16 bit Two's Complement data of each axis is retrieved from the registers of the accelerometer based sensor, ADXL345. The Raspbian Wheezy comes with a pre-installed version of Python 2.7 where the program is written.

The secure sensor node prototype program aims to achieve the following objectives:

- Retrieving sensor data from the data registers of the sensor and processing it to give acceleration in m/s2.
- Encryption of this processed data.

TABLE II. COMPARISON OF HARDWARE REQUIREMENTS OF RC4, TINYSEC AND MINISEC

Parameter	RC4	TinySec/MiniSec
ROM	1494	3798
Clock cycles/byte	62.23	186.54
RAM	1068	276

TABLE III. COMPARISON OF SECURITY ASPECTS OF RC4, TINYSEC AND MINISEC

Parameter	RC4	TinySec	MiniSec
Authenticity	Yes	Yes	Yes
Replay action	Yes	No	Yes
Tolerance to message loss	Yes	No	Yes

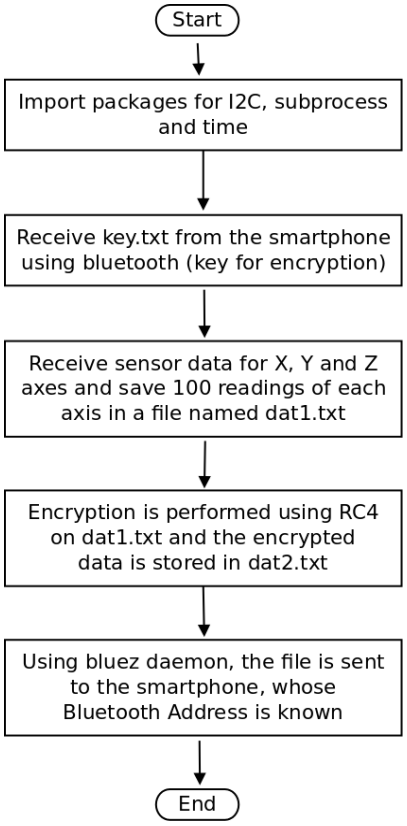

Figure 3. Program flow of the secure sensor node

- Sending this data to the phone using Bluetooth.

The following modules have been imported in the python program to assist in the above processes:

- The 'SMbus' module, known as python-smbus: it is a Python module which allows SMbus access through the I2C/dev interface on Linux hosts.
- The os and the subprocess module: they are used to execute command line instructions in python.

This part of the program describes the process by which data is retrieved and processed from the accelerometer based sensor.

C. Encryption

We have used encrypted Bluetooth transfer from the secure sensor node prototype to the smart phone. This section provides a brief description of the RC4 encryption algorithm used in the program, along with a comparative study between RC4 algorithm and two other encryption algorithms TinySec [6] and MiniSec [7]. Table 3 and Table 4 compare the encryption techniques. The values have been taken from a previous work on comparison of encryption techniques[10]. RC4 is a better technique than the TinySec and MiniSec and hence we have used it in our work.

D. Description of the RC4 Encryption Algorithm

The RC4 algorithm was designed in 1987 by Ron Rivest for RSA Security. RC4 is a variable key size stream cipher. The operations being used in RC4 algorithm are byte oriented. Random permutations are used in this algorithm. Every byte of output requires eight to sixteen operations, and it is obvious to expect that, cipher can run quickly in the software. The RC4 algorithm is remarkably simple and quite easy to explain. The size of key is of variable-length and varies from 1 to 256 bytes (8 to 2048 bits). In RC4 algorithm 256 bytes state vector S is initialized which contains S[0], S[1], …, S[255] elements. Each element of this vector stores 8 bit value which varies from 0 to 255. These values are different permutations of 8-bit. In vector S, values are stored in symmetric fashion. RC4 algorithm includes key scheduling mechanism to per-mutate different values of elements of vector S and pseudo random generation mechanism to generate a byte 'k' from vector S. Generation of each value of k leads to the permutations of entries of S once again.

This section is responsible for encryption of all the sensor values present in the file 'dat1.txt' and storing them in the file 'dat2.txt'. A file, 'key.txt' is received from the mobile phone using the Bluetooth module connected to the Raspberry Pi, right before the encryption code is implemented. This file accomplishes two tasks:

- Implementing the RC4 algorithm to encrypt the sensor data present in 'dat1.txt'.
- Decrypting the encrypted data (present in the file 'dat2.txt') in the mobile phone, after receiving it from the Raspberry Pi over Bluetooth.

The encryption code is compiled using a GCC compiler. Since the encryption code is written in C and the main part of the code (sensor interfacing) is written in python, implementation of the command line compilation as well as the execution of the encryption code is done in the python program using subprocess module.

E. Bluetooth Interfacing and Results

Bluetooth is a wireless technology (IEEE 802.15.1) used to exchange data over short distances (using short-wavelength radio transmissions in the ISM band from 2400–2480 MHz). Bluetooth is used in the secure sensor node prototype to perform two major tasks:

- Accepting the 'key.txt' file from a mobile phone.
- Sending the file, 'dat2.txt'containing the encrypted data to a mobile phone whose MAC address is hard coded in the program.

Bluez package has been used in the design of this secure sensor prototype, which contains the Bluetooth protocol stack for Linux. A Bluetooth USB dongle mentioned above is connected to one of the USB ports of the Raspberry Pi to establish a connection with a mobile phone. An Android's core Bluetooth code – BluetoothShare.java has been implemented in the mobile phone application to make our custom SPP connection.

Obexftp has been used for the transfer of files using Bluez. It is implemented as a collection of command line instructions in the python program by using os and subprocess modules. Different mobiles have different data channel numbers (OPush channel number). The program ends with the deletion of the files 'key.txt', 'dat1.txt' and 'dat2.txt'. This is done so that for every fresh run of the program, real time values are received from the accelerometer based sensor. Command line instructions are implemented in the python program to complete the above process of deletion.

The Raspberry Pi runs a python script which awaits the secret key to be shared by the mobile device in order to start collecting data from the sensor. The mobile device, which is an Android device in our case, runs an application. This application sends the secret key to Raspberry Pi over Bluetooth and then waits to receive encrypted data from the Raspberry Pi. The Raspberry Pi, upon receiving the secret key, starts collecting sensor data and after a stipulated time, encrypts it using the key and sends it over Bluetooth to the mobile device. The mobile device receives the encrypted data and then decrypts it for further processing.

Fig. 5(a) shows the secret key stored on the mobile device used for secure communication with the Raspberry Pi over Bluetooth. Fig. 5(b) shows the screen when the secret key is shared with Raspberry Pi. Fig. 5(c) shows the screen when the application is receiving encrypted data. Fig. 5(d) shows the encrypted file whereas Fig. 5(e) shows the decrypted file containing the sensor data. The 16 bit Two's complement data from the accelerometer data register has been stored in a double data type variable, to prevent loss in accuracy of data, after multiplying the suitable sensitivity ratio and gains to the raw data. Hence the data as seen in the screen snapshots of the prototype is to the 12-13 decimal places. The data before encryption and after decryption match to the last digit in the decimal. Hence the 12-13 decimal places in the sensor data is due to the initial processing of the sensor data. These are the actual readings on the prototype. An actual sensor would have shorter readings.

IV. RELATED WORK

Mobile phones are extremely vulnerable to software-based attacks and require secure communication with sensor nodes. The Plug-n-Trust module [3], where a mobile phone is

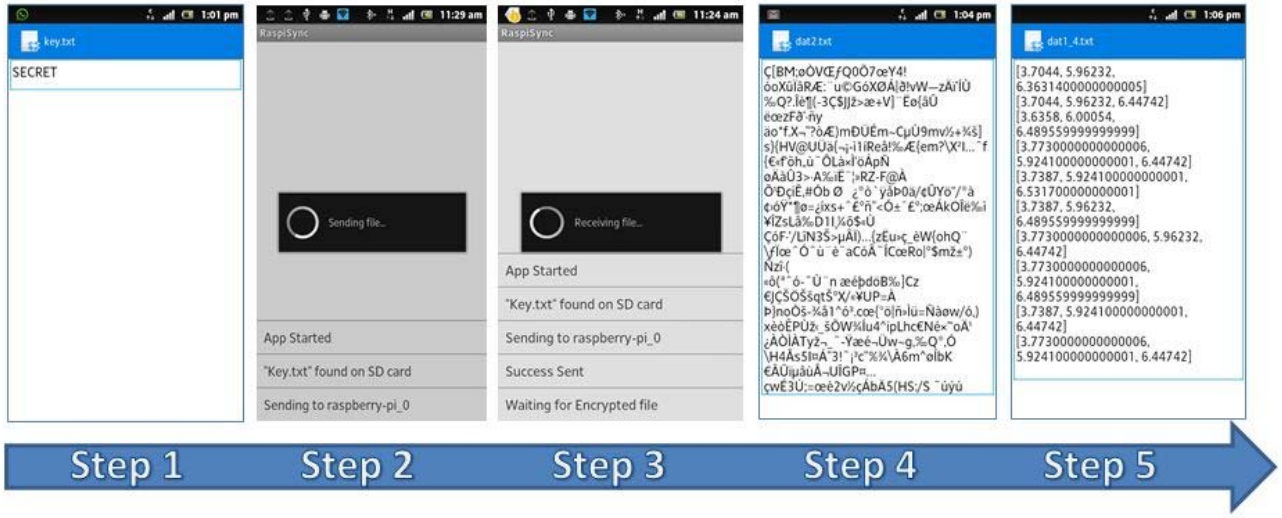

Figure 5. Screenshots of Android application running on the smart phone: (a) Secret key stored securely on smart phone (b) App sends the key to raspberry pi live at the time of encryption via Bluetooth (c) App waits for the encrypted file having body sensor data (d) Received encrypted file (e) App decrypts the data received using RC4 with same key and decrypted data is shown to the user.

responsible to collect the data from the various sensors (connected to the body) suffers from security issues [8].The secure sensor node prototype designed with the Raspberry Pi guarantees more security as compared to Plug-n-Trust module:

- Wired communication occurs between the Raspberry Pi and the sensor so there is no fear of information loss or security.
- The processing and encryption is done within the Raspberry Pi using RC4 encryption which is much more secure as compared to AES which is implemented in Plug-n-Trust module[10].

V. CONCLUSION AND FUTURE WORK

In this work, we have designed a secure sensor node prototype at a low cost. The Raspberry Pi is the best choice because of its high performance/cost, good memory capacity and being the cheapest single board computer available in the market as discussed in section II. The size of the Raspberry Pi being that of a credit card provides compactness and portability to the sensor node. The secure sensor node prototype has been designed using an accelerometer based sensor which gives the x, y and z readings, which can be used for fall detection in body sensors for elderly people. The prototype is unique for using a secure encryption algorithm RC4 as discussed in section III. The Raspberry Pi based prototype has interfaces which are easy to use, and can help designers in investigation and development of new sensors as well as encryption and compression algorithms for future sensors. This secure sensor node prototype can be used to develop a body sensor module which involves processing, encryption and sending of large amount of data gathered from multiple sensors. The processor needs to have high processing ability and also has a large memory to handle many sensors and perform all the operations in real time without significant delay. Hence, Raspberry Pi being the cheapest and one of the smallest SBC available in the market is the best choice for designing a prototype for body sensor module.

We plan to improve upon the design of the prototype by making it a battery operated with switches to start the sensor. The prototype can be used to implement other sensors and to investigate other encryption techniques along with new interfaces like Bluetooth low energy and NFC.

REFERENCES

[1] Jay Chen,Karric Kwong,Dennis Chang, Jerry Luk,Ruzena Bajcsy, “Wearable Sensors For Reliable Fall Detection”,Engineering in Medicine and Biology 27th Annual Conference(IEEE) Shanghai China,September 1-4, 2005.

[2] “World’s elderly to overtake number of infants”,an article in the The Telegraph, UK,18th June ,2013.

[3] Plug-n-Trust: Practical Trusted Sensing for mHealth .Jacob Sorber, Minho Shiny, Ron Peterson, David Kotz,Institute for Security, Technology, and Society, Dartmouth College, Hanover, NH, USA Dept. of Computer Engineering, Myongji University, South Korea

[4] Datasheet archives, contains datasheet of various ICs(ADXL345), www.datasheetarchive.com.

[5] S k Pang Electronics, Regarding the interfacing of sensors with the raspberry pi, www.skpang.co.uk

[6] Chris Karlof, Naveen Sastry, David Wagner, “TinySec: A Link Layer Security Architecture for Wireless Sensor Networks”, In ACM, November 2004.

[7] Mark Luk, Ghita Mezzour, Adrian Perrig, Virgil Gligor, ”MiniSec: A Secure Sensor Network Communication Architecture”, In ACM, April 2007.

[8] Tassos Dimitriou, Krontiris Ioannis, “Security Issues in Biomedical Wireless Sensor Networks”, Applied Sciences on Biomedical and Communication Technologies First International Symposium, conference publication, 2008.

[9] Getting started with Raspberry Pi ,Matt Richardson and Shawn Wallace ,published by O’Reilly Media, First release December 2012.

[10] Shervin Amini, Richard Verhoeven, Johan Lukkien, Shudong Chen, “Toward a Security Model for a Body Sensor Platform”, IEEE International Conference on Consumer Electronics (ICCE), 2011.