Imperial Journal of Interdisciplinary Research (IJIR)
Vol-3, Issue-2, 2017
ISSN: 2454-1362, http://www.onlinejournal.in

Reduction of IoT Communication Barrier Problem Using Smart Phone SDRs

N.Keerthanan[1] **&** M.Jothimani[2]
PG Student[1], Assistant Professor[2]

Abstract: *The smartphones have become powerful enough to process software defined radio (SDR) for some known wireless protocols. Moreover, we show that the SDRs can be packaged as "apps" and be downloaded from app stores for OS-independent deployment. we implement a prototype architecture that has all the SDR logic and supporting middleware on an Android smartphone which uses a USRP as the simple RF-end.We proposed SDR with cryptography based security mechanism to security algorithms like Blowfish Security Algorithm . Blowfish is known for both its tremendous speed and overall effectiveness as many claim that it has never been defeated. Improving encryption and decryption aspects of the algorithm, which is already exist and creates the way for an excellent security. Energy-aware routing algorithms to be proposed for Wireless Sensor networks, called reliable minimum Hybrid Dynamic Energy Routing Protocol (HDERP).HDERP addresses important requirements of WSN for reducing high power consumption. We demonstrate IEEE 802.11p and IEEE 802.15.4 SDRs on a smartphone respectively communicate with a ZigBee sensor mote, a ZigBee, smart light bulb and commercial Wireless Access in Vehicular Environment (WAVE) device concurrently.*

Index Terms*—Internet-of-Things (IoT), software-defined radio (SDR), smartphone. implementation, IEEE 802.11p, IEEE 802.15.4.*

1.Introduction

The future Internet, designed as an "Internet of Things" is foreseen to be a world- wide network of interconnected objects uniquely addressable, based on standard communication protocols". Identified by a unique address, any object including computers, sensors, RFID tags or mobile phones will be able to dynamically join the network, collaborate and cooperate efficiently to achieve different tasks. Including WSNs in such a scenario will open new perspectives. Covering a wide application field, WSNs can play an important role by collecting surrounding context and environment information. However, deploying WSNs configured to access the Internet raises novel challenges, which need to be tackled before taking advantage of the many benefits of such integration.The main contributions of this paper can be summarized as follows: We look at WSNs and the Internet holistically, in line with the vision where WSNs will be a part of an Internet of Things. Smartphones are frequently used for control and interaction, and will continue to play a central role in future IoT environments. Unfortunately, however, due to limited *modi operandi*, energy constraints, and various other reasons, many IoT devices may use mission-tailored or proprietary wireless protocols that smartphones do not natively speak. Consequently, there will likely be a 0"language barrier" be-tween them. Although third party device aggregation or integration services could provide one solution for control gate-ways, they can create problems such as dependency on their business models and technical support, suboptimal device quality , and limited scalability as new IoT protocols will continue to emerge. (But the existence of these commercial services strongly testifies to the need for supporting multiple IoT protocols that are non-native to smartphones, in future.) The Internet of Things (IoT) is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people

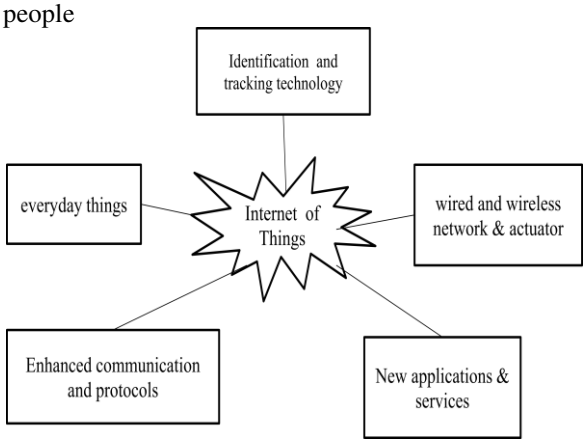

Figure 1 Sensor Network Process

The wide wireless sensor network application field can be divided into three main categories according Monitoring space, monitoring objects and monitoring interactions between objects and space.in this fig 1 shows that classification can be extended by an additional category monitoring human beings. One example of the first category is environmental monitoring. WSNs are deployed in

particular environments including glaciers, forests and mountains in order to gather environmental parameters during long periods. Temperature, moisture or light sensor readings allow analyzing environmental phenomena, such as the influence of climate change on rock fall in permafrost areas . The second category centers on observing particular objects. Structural monitoring is one of the possible illustrations of this category. By sensing modes of vibration, acoustic emissions and responses to stimuli, mechanical modifications of bridges or buildings indicating potential breakages of the structure may be detected. Monitoring interaction between objects and space is the combination of both previous categories and includes monitoring environmental threats like floods and volcanic activities. Presenting an extension to the presented classification, the last category focuses on monitoring human beings. Worn close to the body, the deployed sensors can gather acceleration information and physiological parameters like heart beat rate. Especially in applications in the medical area, such deployments may help diagnosing bipolar patients and monitoring elderly people in a home care scenario. the fundamental *feasibility* of the proposed idea would be the speed and the power consumption issues of SDR. These are the Achilles' heel of SDR, and we do recognize that they would become only worse on limited capacity devices such as smartphones. Fortunately, that we are targeting the IoT alleviates the issues significantly. First of all, most IoT protocols are not designed for high-speed data exchange, so they lend themselves easily to software implementations. For instance, a smart meter reading protocol ERT works only at 32.768 kbps [11]. But such speed typical of many IoT applications is readily as smartphones are becoming ever more powerful in terms of computing capacity. Indeed, we will demonstrate applications whose wireless protocols the smartphones .

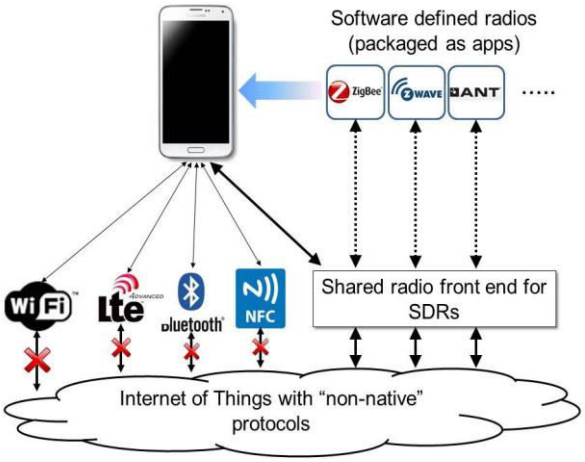

Fig 2 summarizes the proposed concept.

that today's smartphones are already capable of bearing real-time signal processing workloads imposed by low-speed protocols such as IEEE 802.15.4, and even higher-speed protocols such as IEEE 802.11p at up to 6 Mb/s (Section 6.2). Fig. 2 offers a perspective on smart-phones' potential as a SDR platform. It shows the trend in the clock rate of the application processor (AP) and the number of cores in top-of-the-line smartphones from some well-known vendors. We can observe that the AP clock rate is steadily increasing, and some have reached 2.5GHz. Even when the raw clock rate decreases, the processing capacity is effectively increased through increased cores or word width. IoT communication impose much less stringent constraints. For example, one would not turn on and off a IoT lightbulb or a door lock all day long. These command type interactions rarely require extensive data exchange. Even in background sensing such as a fire alarm that requires continual monitoring, duty cycling is usually employed. Such considerations are espe-cially important as smartphones operated on limited battery power. For example, in our unoptimized implementation of smartphone SDR that needs to drive a large external radio front board, we will show the continual operation for ZigBee communication draws close to 400 mA (Section 6.3). Considering 3000 mAh battery for the most recent smartphones such as Galaxy S7, this means less than 8 hours of operation is possible. However, even a 10% duty cycling will extend it to a much longer operation lifetime than a typical recharge period for smartphones. In summary, the IoT environment provides a unique niche for smarphone SDR that makes it viable and valuable. the proposed smartphone SDR concept is feasible by running on smartphones

2.Background And Related Work

Software defined radios (SDRs) have come through two generations of platforms so far. Programmable hardwares such as field programmable gate arrays (FPGAs) were used in the first generation . As personal computers (PCs) became powerful enough to provide the computation capability required for real-time signal processing, they opened the second generation . Owing to the long history of the technology, there is rich literature on software radio. Here, we briefly discuss those that are directly related with the investigation in this paper. Schmid discusses the implementation experiences of the IEEE 802.15.4 module for the GNU Radio. It focuses on the encoding and decoding methodologies as the

physical (PHY) layer is much more computation-intensive part of the protocol. This work shows that the software implementation on the dual Pentium IV platform with 2.8GHz clock and hyper threading performs close enough to the hardware implementation of the 802.15.4 protocol on the Chipcon CC2420 radio used in various mote platforms. Analyze the workload of the signal processing algo-rithms in the baseband operation of contemporary wireless networks. The workload characterization is done on two levels, system architecture and individual algorithms like modulation and coding. In this Fig 2 discuss the design and implementation of a high-performance SDR for Wi-Fi using general purpose multi-core processors. It identifies quantitative requirements such as bus throughput and computation power as well as constraints for real-time operation, in building such a system. It can run Wi-Fi (802.11a/b/g) protocol and communicate with commercial systems. Bloessl *et al*. [18] implements a IEEE 802.11p GNU Radio stack. Although we do not use the source in our work, we adopted their idea on extracting traffic dump during the communication with the Wireless Access in Vehicular Envi-ronment (WAVE) on-board unit (OBU) for the validation of our prototype. , powerful general-purpose computing platforms such as server class PC [14] or PC with general purpose graphic processing unit (GPGPU) loomed as more flexible alternatives [15]. Either way, the notion of software radio has been geared toward a small number of fixed stations (*e.g.* base stations). Moving to the newest and the most prevalent platform, *i.e.*, billions of smartphones, has only recently been attempted [23], [24]. Moreover, the PC-based SDRs run as part of the OS, but not as an application program. The potential implications of making the SDR as a downloadable application software separate from the OS has not been fully explored. smartphones supporting IoT protocols in SDRs, the solutions will give us valuable lessons on how to schedule the accesses and resolve possible conflicts between multiple IoT protocols, in accessing the radio front.

3.Providing Radio Frontend To Smart Phone Sdrs

Radio frontends (RFs) directly to the application processor (AP), the first technical hurdle in realizing the envisioned concept is providing a RF to IoT SDRs. Although smartphone vendors finally began to offer swappable plug-in hardware modules, there is no plug-in RF module available yet. So for the sake of proof-of-concept, we use an external RF and connect it to a smartphone. The only available wired interface on today's smartphones to connect to the external RF is the Universal Serial Bus (USB) interface

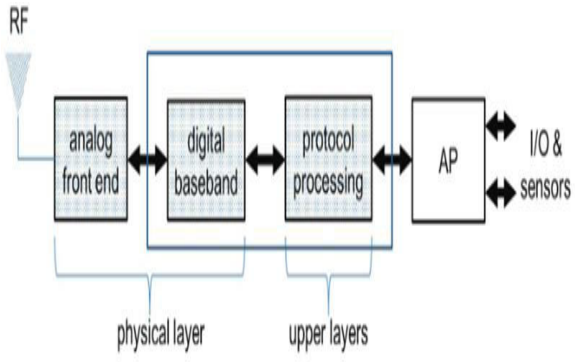

Fig 3 Connecting the AP to an external radio frontend through USB

4 IMPLEMENTING SDR AS SMARTPHONE APPS

SDR work as a smartphone app poses quite a few challenges because we must meet stringent protocol deadlines in the signal processing for frame reception and transmission. Specifically, we must meet the timing require-ments in encoding, decoding, response generation (*e.g.* ACK and CTS), and event trigger timing (*e.g.* CCA) . Other-wise, protocols will not work. In this section, we discuss how we implement the SDR as an app while meeting thetiming requirements as much as possible. The software architecture for our smartphone SDRs. The two shaded blobs comprise the implemented smartphone SDR system. When the SDR app is downloaded from an app store and installed on the smartphone, both blobs are created. In the lower blob, two SDRs exist as embedded libraries whose combined size is 5.47 MB.

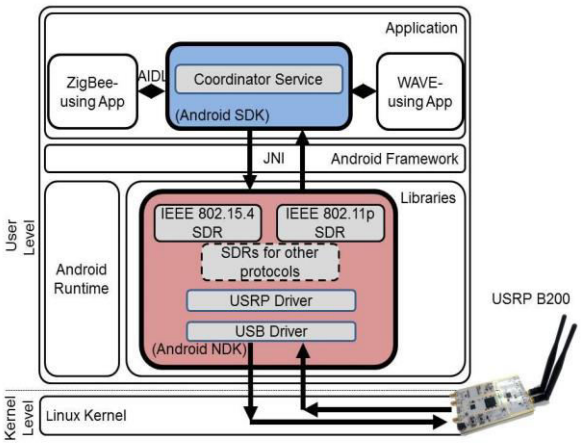

Fig. 4. Software architecture of the coordinator on an Android smart-phone

Available SDR sources originally developed for Intel CPU architecture, and port it to the ARM architecture used on the test smartphones. For the IEEE 802.11p PHY layer is essentially a half-clocked 802.11a PHY, so we port SORA [14] 802.11a SDR code. For the IEEE 802.15.4 SDR, we port the UCLA extension to the GNU Radio [16]. Other IoT wireless protocols can be later added. Notice that we do not modify the kernel. functionalities are implemented on the user level. Since the SDR app is self-contained with nec-essary libraries, it does not require any additional support from Android or the Linux kernel to run on smartphones.

4.1 Language

Android apps, we do not use only Java to implement main SDR routines. The reason is threefold. First, existing SDR sources that we want to reuse are in other languages such as C++ and C, and choosing Java means having to translate the entire existing source to Java. Second, Java is known to be slower than these languages, so using it runs a risk of missing protocol deadlines more frequently. Third and most importantly, the Single Instruction Multiple Data (SIMD) functions are not usable in Java.

4.2 SIMD

SIMD code in SORA are based on Windows Driver Development Kit (DDK) and Intel Streaming SIMD Exten-sions (SSE), we need to port the SIMD instructions to the An-droid NDK and ARM architecture. Unfortunately, the SIMD instructions cannot be mapped one-to-one between the two processor architectures. Arithmetic and logic instructions are mostly simply translated.

5 Performance of SDRs

Implemented the IEEE 802.11p and IEEE 802.15.4 SDRs as apps, and executed them on a few smartphones with different processing capabilities and architectures.

5.1 Smartphone platforms

We mainly use two smartphone platforms in our evaluation: Samsung Galaxy S4 LTE-A and Galaxy S5. Their specifica-tions are given in Table 2. But in order to get a perspective in how the measured performance projects into future, we will provide the data from older phones such as S2 as deemed necessary.

5.2 Execution times

IEEE 802.11p SDR execution time In this experiment, we perform IEEE 802.11p packet transmission and reception using the SDR. As in SORA [14], we use a separate thread for the Viterbi decoder, as it is known to be the heaviest workload in the baseband processing on PC testbeds [14], [15]. Fig. 11 compares the total signal processing time for encoding and decoding a 1 KB 802.11p packet on various smartphones.

6.Secure Communication In An Internet Of Things

A key challenge for IoT towards SC applications is ensuring their reliability, incorporating the issues of security, privacy, availability, robustness and flexibility to changing environmental conditions.

6.1 Data Integrity

Data integrity checking protocols aim to maintain the complete structure of stored data, ensure its correctness and protect this data from lost or corruption keeping in mind economies of scale, practicality and support for dynamic data operations. During the past decade severalprotocols were designed to achieve data integrity in cloud computing. These protocols use either encryption techniques. Due to the similarities between the IoT and cloud computing when it comes to data storage, privacy and confidentiality (i.e., both systems need to maintain the integrity of growing amounts of data stored on remote servers where the data is frequently modified), the same protocols adopted for data integrity in cloud will meet the requirements of the IoT. Affording secure and efficient big data aggregation methods is very attractive in the field of wireless sensor networks research. In real settings, the wireless sensor networks have been broadly applied, such as target tracking and environment remote monitoring. However, data can be easily compromised by a vast of attacks, such as data interception and data tampering, etc. In this paper, we mainly focus on data integrity protection, give an identity-based aggregate signature scheme with a designated ver-ifier for wireless sensor networks.

6.2 ID based cryptography

In an ID-based cryptography, the user's public key is any publicly known and unique identity information, such as the serial number, and the user no longer needs a certificate to prove its identity. ID-based aggregate signature schemes have been presented . Up to now, a great many aggregate signature schemes have rapidly emerged in various settings, such as in PKIs in Certificateless Public Key Cryptography (CL-PKC), respectively. show security drawbacks of the certificateless aggregate

signature scheme by demonstrating some kinds of attacks.It will mainly focus on designing the aggregate signature scheme which can resist coalition attacks.Aggregator is a special sensor node with a certain ability to calculation and communication range. Data integrity checking protocols aim to maintain the complete structure of stored data, ensure its correctness and protect this data from lost or corruption keeping in mind economies of scale, practicality and support for dynamic data operations. Sensor node has limited resources in terms of compu-tation, memory and battery power.

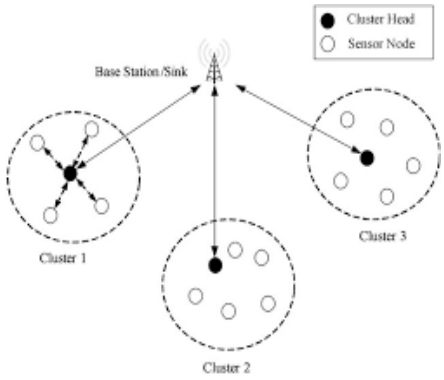

Fig 5 cluster based network

6.3 Communication barrier(Data integration) problem in Internet of Things

Data integration is front and center with most *IoT* strategies that developed, considering that the ability to move and process *data* is really the main *problem* to solve. ... Emerging technology such as *IoT*, cloud computing, and even big *data* typically places *data integration* into the afterthought category. Encryption and decryption algorithms should be very efficient process to reduce the malicious node occurred in the IoT communication . To improve the security process when the hackers should be hack the information between banker (smart phone SDRs user) and IoT device.

6.4 Blowfish algorithm used for Internet of Things

Blowfish algorithm (BA) is a symmetric block cipher with a 64-bit block size and variable key lengths from 32 bits up to a maximum of 448 bits. In order to measure the degree of security of blowfish algorithm.

6.4.1 Collaborative Attacks

Collaborative attacks (CA) occur when more than oneattacker or running process synchronize their actions to disturb a target network

Denial-of-Messages (DoM) attacks

- Malicious nodes may prevent other honest ones from receiving broadcast messages by interfering with their radio signal.

Blackhole attacks

- A node transmits a malicious broadcast informing that it has the shortest and most current path to the destination aiming to intercept messages.

Wormhole attacks

- An attacker records packets (or bits) at one location in the network, tunnels them to another location, and retransmits them into the network at that location

Denial-of-Messages (DoM) attacks

- Malicious nodes may prevent other honest ones from receiving broadcast messages by interfering with their radio signal

Rushing attacks

- An attacker disseminates a malicious control messages fast enough to block legitimate messages that arrive later(uses the fact tha only the first message received by anode is used, preventing loops)

Malicious flooding

- A bad node floods the network or a specific target node with data or control messages.

As **Internet of Things** (**IoT**) can be broadly used in many fields, the security of **IoT** is gaining importance. In **IoT**, all the devices are connected. Internet of Things (IoT) can be broadly used in many fields, the security of IoT is gaining importance. In IoT, all the devices are connected. If a hacker manages to enter the network, he may access confidential datas. So for IoT, an information transmission security mechanism is essential in addition to the authentication mechanism. Beginning with the concept of IoT, its architecture and security issues, this paper analyzes various security mechanisms for IoT and the significance of cryptography in IoT. An efficient cryptographic algorithm "Blowfish" is selected based on several comparisons.Blowfish is a perfect algorithm for Internet of Things.

7 Power consumption

The power consumption of our prototype system. Since the smartphones are battery-operated, power consumption is an important aspect that may affect the viability of the smartphone SDR. In IoT applications, how-ever, the interactions between smartphones and IoT devices will typically be ephemeral and sporadic. For example, turning lights on and off, or checking the room temperature using a smartphone will be done in a couple of seconds. For Internet of Things,

the power consumption of the software defined radio will not be the biggest concern. the phone vendors support IoT SDRs in future, they will provide an internal RF that will consume much less power than the external USRP B200 board that we use for our prototype. So what we present here is for the qualitative understanding of power consumption dynamics on a smartphone-based SDR, and the numbers should not be taken as representing the power requirement for the smartphone SDR systems in future.

7.1 Measurement

Using Monsoon power monitor, we observe the battery drain during at least one second of operation in each case, and the results are shown in Fig. 13. The input voltage is 3.85 V in Galaxy S5, so power consumption can be obtained by multiplying the value to the current draw of each case. Case 1 is when there is no communication, as the USRP B200.

The current draw in the first case thus represents power consumption mostly by display. By comparing Cases 1 and 2 in Fig. 13, we observe that the USRP B200 draws approximately 150 mA just by connecting. For all cases except Case 1, this baseline current draw is always incurred by B200. Now let us estimate additional current draws. For Case 3, we load the images and configure the operational parameters, getting the USRP ready to be used. This signifi-cantly increases the current draw by approximately 500 mA, Then in Case 4, we turn on the sensor mote and let it transmit the sensory data towards the smartphone. It adds another 300 mA to the current draw. On the other hand, comparing Cases 4 and 6, we notice that IEEE 802.11p reception (with no packet present) consumes more battery than the IEEE 802.15.4 decoding. It adds approximately 430 mA. When the WAVE beacons begin arrive and full decoding logic is engaged, the additional current draw exceeds 500 mA.

7.2 Test configurations

First, it can be disconnected from the USB port of the smartphone (Case 1). Second, it can be connected to the USB port, but the firmware and FPGA images and the parameters are not configured at B200 (Case 2). So in this state the external RF is not usable,

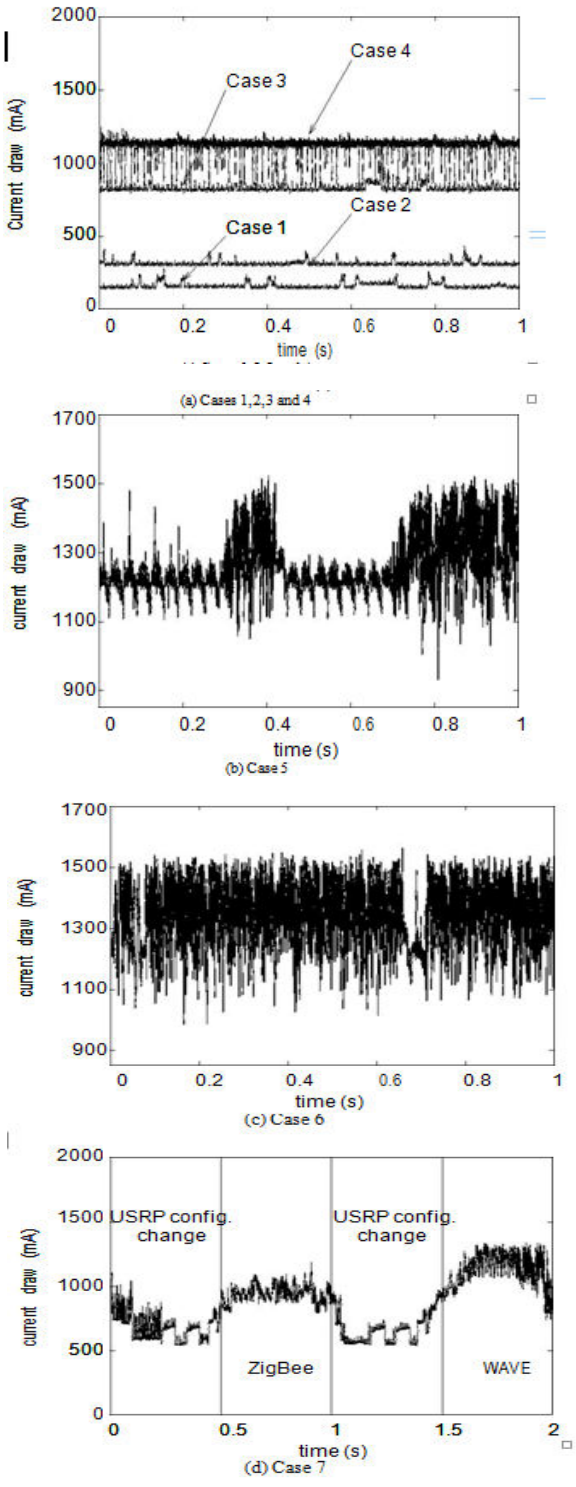

Fig. 6. Current draws for the test configurations

8.Conclusion

The implementing software defined radio on smartphones as an IoT device. On feasibility, we demon-strate that today's smartphones are capable in terms of computation capacity to bear the signal processing work-load required to run some popular

wireless protocols. Their hardware architectures that do not consider the possibility of smartphone SDR and block the datapath from RF to application processor is the only remaining major roadblock. Smartphones interact with commercial IoT devices such as a Tmote Skey sensor mote, a Phillips Hue lightbulb, and an ARADA WAVE OBU. With an open and shared RF, smartphones can support less popular protocols, proprietary protocols, and experimental protocols without risking hardware investments. proposed SDR with cryptography based security mechanism to security algorithms like Blowfish Security Algorithm . Blowfish is known for both its tremendous speed and overall effectiveness as many claim that it has never been defeated. Improving encryption and decryption aspects of the algorithm,The energy efficiency of a virtual topology derived from an hybrid dynamic energy routing protocol(HDERP) algorithm for situation awareness to reduce the power consumption of IoT .To focus on monitoring capability a signature based intrusion detection system should be similarly bench marked against a random poer based- deployment. Energy was factored into traffic load cost in the IoT indicator.

9.Acknowledgement

The authors acknowledge the contributions of the students, faculty of KSR College of Engineering,Tiruchengode.,for helping in the design and for tool support. The authors also thank the anonymous reviewers for their thoughtful comments that helped to improve this paper**.**

10.References

[1]Audrey Ann Gendreau, "Situation Awareness Measurement Enhanced for Efficient Monitoring in the Internet of Things", *Computer Science & Computer Information Systems IEEE Region 10 Symposium* 2010.

[2]Cheng Tan and Qingbing Ji, "An Approach to Identifying Crptographic Algorithm from Ciphertext" *IEEE international conference on communication software and networks* 2016.

[3]Dejana and ugrenovic,"CoAp protocol for Web based monitoring in IoT healthcare application" *23th telecommunication forum* 2015.

[4]Dongyu Wang, Dixon Lo, Janak Bhimani , Kazunori Sugiura "AnyControl IoT based Home Appliances Monitoring and Controlling" *IEEE 39th Annual International Computers, Software & Applications Conference* 2015.

[5]J. Kulik, W. R. Heinzelman, and H. Balakrishnan, "Adaptive protocols for information dissemination in wireless sensor networks," in *Proc. 5th Annual ACM/IEEE transaction. Conf. Mobile Computer Network, Seattle,WA, USA,* pp. 174–185,1999.

[6]June kim and Seun Hyeon, "Implementation of an SDR system using graphics processing unit" topic in Radio communication,,*IEEE communication magazine* 2010.

[7]Kuen-Min Lee ,Wei-Guang Teng, "Ponit –n-Press: An Itelligent Universal Remote Control

[13]S. Chen, S. Tang, M. Huang, and Y. Wang, "Capacity of data collection in arbitrary wireless sensor networks," *IEEE Transaction. Parallel* System vol. 23, no. 1, pp. 52–60, 2012.

[14]K. Akkaya and M. Younis, "A survey on routing protocols for wireless sensor networks," *Ad Hoc Netw.*, vol. 3, no. 3, pp. 325–349, 2005.

[15]K.-W. Fan, S. Liu, and P. Sinha, "Structure-free data aggregation in sensor networks," *IEEE Transaction Mobile Computer* vol. 6, no. 8,pp. 929–942, 2007.

[16]Q.Mamun, "A qualitative comparison of different logical topologies for wireless sensor networks.

wireless Energy Transfer"*IEEE Transaction on mobile computing* 2016.

[12]W.B.Heinzelman, A. P. Chandrasekasan, and H. Balakrishnan, "An application specific protocol architecture for wireless microsensor networks,"*IEEE Transaction Wireless Communication* vol. 1, no. 4, pp. 660–670, Oct. 2002.

[8]Yongtae Park and Seungho Kuk, Inhye Kang, Hyogon Kim, "Overcoming IoT Language Barriers Using Smartphone SDRs," *IEEE Transactions on Mobile Computing 2012*.

[9]Younggi Kim ,Younghee Lee. "Automatic Generation of Social Relationships between Internet of Things in Smart Home using SDN-based Home Cloud"*29th International Conference on Advanced Information Networking and Applications Workshops on* 2015.

[10]Yuanyu Zhang, Yulong Shen, Hua Wang, Jianming Yong "On Secure Wireless communications for IoT under Evesdropper collusion"*IEEE transaction on Automation science and Engineering* 2015.

[11]Wenzheng Xu,Weifa Liang. "maximizing charging satisfication of smart phone users Via