Journal of Telecommunications System & Management

Iwendi and Offor, J Telecommun Syst Manage 2012, 2:1
DOI: 10.4172/2167-0919.1000106

Research Article Open Access

Alternative Protocol Implementation for Wireless Sensor Network Nodes

C. O. Iwendi[1]* and K. J. Offor[2]

[1]School of Engineering, University of Aberdeen, Aberdeen, Scotland, UK
[2]Department of Electrical and Electronic Engineering, Anambra State University, Uli, Nigeria.

Abstract

The nodes being used in Wireless Sensor Networks (WSN) pose particular challenge to communicate with one another using TCP/IP especially when considering security applications. This paper investigates the limitations of current TCP/IP structure and an integration using micro IP (µIP) and proposes an enhanced security technique for WSN nodes to solve issues arising from the node congestion due to data distribution of keys with a better method for securing the data in the network. The summation of this method involves the integration of µIP stack to key management scheme, allowing the network to have extra security in the transmission and reception of data, and can be applied in hazardous areas where the sensor nodes are used without complications.

Keywords: Blom; AES; Pairwise key; WSN; Preallocation; Pre-distribution; Contiki; Cooja; µIP; Smart performance

Introduction

Wireless Sensor Networks (WSN) are widely used by diverse communication systems [1] and the effective use of the Internet Protocol technology in WSNs is a key prerequisite for the realization of the Internet of Things vision [2]. Therefore, with a high level of objectivity, advances in WSN technology has gradually turned into large scale utilization. The challenge is to enable networks and systems the possibility to self manage their complexity and communicate easily. The self-powered WSN shows a facility with sensors and a capacity to communicate over TCP/IP in a centralized building automation system. This is an example of implementation of the internet of things.

The µIP is an open source TCP/IP stack system that is capable of being used with microcontrollers and sensor nodes [3]. The goal is to create an interface with the sensor nodes that will have an IP address, simple connectivity and an increased performance. The summary of the reasons for considering this alternative method for security in a WSN setup as discussed in [4,5], is due to the following drawbacks in the non-IP configuration:

a) Limited Processing Power of the Sensor nodes

b) Prone to Failure due to Dynamic topology

c) Energy Consumption Constraint

d) Node Congestion due to data processing on the node itself

e) Large Scale Deployment

This mechanism is aimed at areas where it is difficult to convey information from one node to another. This method integrates the sensor nodes with an IP address and allows the nodes to communicate with the base station using a gateway router as an interface.

Requirements for µIP-WSN

The requirements include the following:

Energy-saving routing: The energy consumption of transmission must be minimised because IP networks are not energy conserving. Also, routing in IP-WSN must be energy saving. Short distances implies less energy to transmit and to route around intruders including poor quality links should be a strong requirement for IP-WSN network [6]

Highly constrained devices: Integration of highly resource constrained devices over IP has been difficult [7]. IP-WSN can be used to achieve network layer interoperability if properly integrated with small error rate and a limited buffer.

Harsh dynamic environments: IP-WSN should have variable link qualities, link/nodes failure at rate higher than the internet [8]. The fact that IP platform relies on an intermediate Web server implies that the Web client does not access the WSN transparently. Lack of transparency increases the complexity of the IP-WSN application development and hampers interoperability and scalability. The gateway should therefore offer proxy functionality to solve this harsh dynamic environmental problem.

Power management: It has been suggested to have WSN sleep mode most of the time in order to conserve power [9]. Therefore, different energy techniques such as energy harvester need to be considered in IP-WSN communications. The type of IP-WSN application will depend on the density of the network and possibilities of reconfiguration. The main goal should include having power as low as possible and that are appropriate for various types of application.

Multi-path and data aggregation: There is a need to have a protocol based on interactions between individual network nodes with a structured traffic patterns and different TCP sections to support multicast topology [10]. This method should encourage data aggregation and dissemination. It will also match the application requirements for a large distributed sensor networks. Load balancing can then be implemented through the multipath along a dynamically computed path to a link to increase the life time of the sensor node collaborating in the routing process.

Heterogeneous capabilities: The presence of heterogeneous capabilities of wireless sensor nodes in an IP-WSN network is predicted to increase network reliability and lifetime [11]. For instance, nodes deployed with an enhanced energy capacity tend to have better performance in networks of different sizes and densities.

µIP limitations and overhead cost

Internet protocol has a great number of limitations and this also increases the overhead cost of using my scheme. They include:

***Corresponding author:** Celestine Iwendi, School of Engineering, University of Aberdeen, Fraser Noble Building, AB24 3UE Aberdeen, Scotland, E-mail: ciwendi@abdn.ac.uk

Received May 17 2013; **Accepted** June 24, 2013; **Published** June 26, 2013

Citation: Iwendi CO, Offor KJ (2013) Alternative Protocol Implementation for Wireless Sensor Network Nodes. J Telecommun Syst Manage 2: 106. doi:10.4172/2167-0919.1000106

Managing complexity: Complexity in IP and µIP is inevitable due to difficult-to-accomplish integrations with products and workflows, and the flow of security vulnerabilities orchestrated by architectural, technical and design- related applications limits the merits of IP.

Lack of end-to-end routing consistency: IP protocols have favoured how quickly a network reacts over consistency. Responsiveness has come at the cost of routing loops. This has become a bigger issue with Internet routing [12]. For instance, when a protocol behaviour becomes complex and cannot be predicted, the routers tends to operate by forwarding packets along loops with no indication of when the network will converge to a stable state. A stable node will ensure that a router is adopted only after all dependent routers have agreed upon a consistent view of global state.

Multi-topology routing: This is the routing that allows each router in a network to maintain several valid routes to each destination [13], thereby increasing the possibilities of spreading traffic towards a destination over multiple paths with connectionless routing proposals. The shortcoming of current IP is based on the inability to change the routing in response to traffic dynamics without triggering an Interior Gateway Protocol IGP re-convergence.

Data transaction reliability: The reliability and fault tolerance for Internet applications is currently limited. This is due to the lack of a common protocol by which a distributed node synchronises thereby leading to unreliable and inconsistent flow of information in distributed systems. For instance, there is a higher percentage of overhead generated due to the header bytes transmitted with TCP packets of data payload [14].

Application-aware limitation and multi-layer recovery: Current IP and the Transport networks do not interact. This is a fact that has lead to several consequences for both networks because IP is based on packet switching, which guarantees that Internet core cannot benefit from more scalable circuit switches or dynamic circuit switching [15]. For instance, a dynamic circuit can recover faster from failures, provide bandwidth-on-demand, or guarantee low-latency but it is very difficult in today's packet-only switching.

Network implementation essentials–alternative protocol implementation

We have implemented the Enhanced pairwise key pre-distribution scheme on two networking protocols, µIP and RIME with a view to compare the performances, energy efficiency and efficiency in general of the two methods.

Contiki's Rime networking stacks provide an energetically inexpensive way of distributing small amounts of data amongst sensor neighbours. It comes with various functions to send data using unicast or multicast modes by singlehop or multihop schemes. It does not provide any functionality to communicate on Internet Protocol network.

The µIP is an open source TCP/IP stack capable of being used with tiny 8-bit and 16-bit microcontrollers [3]. It implements RFC-compliant IPv4, IPv6, TCP and UDP (the latter two compatible with IPv4 and IPv6). µIP is very optimized, only the required features are implemented. For instance there is a single buffer for the whole stack, used for received packets as well as for those packets to send.

Contiki's µIP networking stack requires an initialisation of various modules required for correct communication in wireless sensor network such as µIP-fw (µIP forwarding) and µIP-over-mesh (µIP over Rime's mesh) modules.

We have implemented the same module used in the Enhanced pairwise key pre-distribution scheme in the µIP to add the ability to communicate on Internet Protocol network. It is expected to have the same or even increased power consumption and decreased performance over the example using Rime protocol. Below, we provide a comparative analysis of the two methods in terms of energy efficiency and performance.

µIP-WSN security application

Wireless sensor node can communicate using the protocol of most networking infrastructure as outlined by the Contiki developers with the help of TCP/IP, although the mechanism is not implemented with sensor deployment. In this section, we shall demonstrate how enhanced security mechanism can be incorporated with µIP for better performance and ease of transmission. This method which involves a modification of µIP open source stack to accommodate the sensor nodes code using the key distribution method is an interface with sensor nodes within an IP domain and provides security for the µIP-WSN without creating problems of overhead, and memory capacity. Data can therefore be easily transferred with the actualization and implementation of this contribution.

The µIP stack is built on large transmitting data packets that offer better packet buffer management. Therefore, the micro IP that we have used in this research is designed to further incorporate the minimal set of components necessary for a full TCP/IP stack developed by Adam Dunkel of the Swedish Institute which support a single network interface. The µIP is a resource-constrained embedded device [6] which makes it ideal for WSN applications. Therefore, any processor that has a lesser amount of memory can support the entire network stack on board, reducing the added expense of external RAM.

In our design, µIP uses a single global buffer for incoming and outgoing packets; this allows application to first process the incoming packet before the next packet arrives, and the application can copy the data into its own buffer for later processing. The Enhanced µIP stack also requires the application to assist in packet retransmission by sending an event to the application. The stack was implemented with enhanced security mechanism making use of a µIP-over-mesh module by default and uses it for both single-hop and multi-hop communication.

The program uses µIP with each sensor nodes having an individually assigned IP address beginning by default at 172.16.0.0 and netmask 255.255.0.0 (routing prefix of an address written in a form identical to that of the address itself). The program runs similarly to the RIME implementation that was discussed [5]. The networks is first compiled and run at the base station so that the sensor nodes can be assigned with 172.16.1.0 IP address and then creating more nodes with node.c compilation and execution. A button click on the node will cause pre-allocation of the pairwise keys while every other click will cause encrypting message which is sent to ID3 with IP address 172.16.3.0. Figure 1 shows the overview of the individual nodes that make up the entire sensor network displayed before the pre-allocation of nodes at the base station. Figure 2 shows the overview of the individual nodes that make up the entire sensor network displayed after pre-allocation with their IP address. Figure 3 shows the implementation and combination of enhanced security for WSN and µIP.

Results / Performance of Different Attacks on Security Scheme

We tested our scheme using Cooja against the network attacks. A software program containing the denial-of-service (DoS) attack was

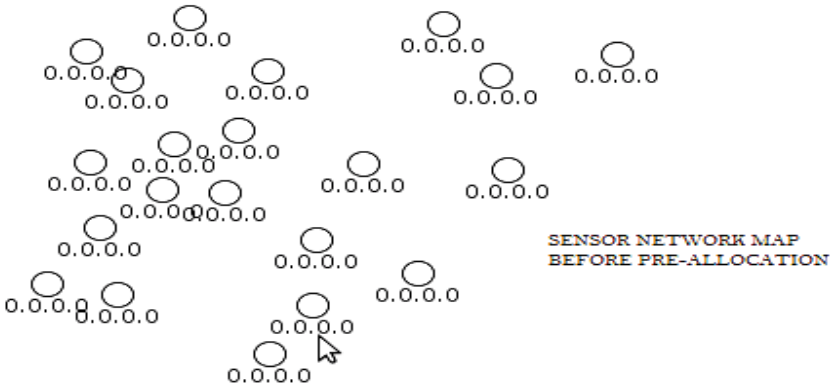

Figure 1: Wireless Sensor Network nodes.

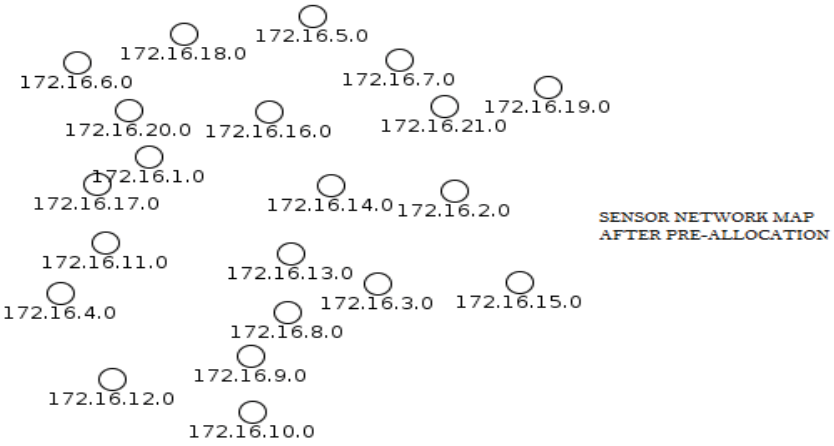

Figure 2: Wireless Sensor Network nodes with IP configuration.

Time ms	Mote	Me
884	ID:13	Contiki 2.5 started. Node id is set to 13.
884	ID:13	Rime address: 13.0
884	ID:13	MAC nullmac RDC nullrdc NETWORK Rime
884	ID:13	IPv4 address: 172.16.13.0
884	ID:13	Starting 'Receive key'
899	ID:3	Contiki 2.5 started. Node id is set to 3.
899	ID:3	Rime address: 3.0
899	ID:3	MAC nullmac RDC nullrdc NETWORK Rime
899	ID:3	IPv4 address: 172.16.3.0
899	ID:3	Starting 'Receive key'
906	ID:21	Contiki 2.5 started. Node id is set to 21.
906	ID:21	Rime address: 21.0
906	ID:21	MAC nullmac RDC nullrdc NETWORK Rime
906	ID:21	IPv4 address: 172.16.21.0
906	ID:21	Starting 'Receive key'
79631152	ID:3	Preallocation is complete
83508805	ID:14	Preallocation is complete
97617235	ID:5	Preallocation is complete
100549305	ID:16	Preallocation is complete
104985324	ID:7	Preallocation is complete
113615974	ID:4	Preallocation is complete
120684230	ID:16	Sending decrypted = Hello, encrypted = □□s!□□□0&C□□r'□Y
120684235	ID:3	Received encrypted: □□s!□□□0&C□□r'□Y□, decrypted: Hello
128784416	ID:6	Preallocation is complete
134784071	ID:7	Sending decrypted = Hello, encrypted = v□f□□ □&□□5 □□□
134784076	ID:3	Received encrypted: v□f□□ □&□□5 □□□□□7, decrypted: Hello
142498730	ID:16	Sending decrypted = Hello, encrypted = □□s!□□□0&C□□r'□Y
142498735	ID:3	Received encrypted: □□s!□□□0&C□□r'□Y□, decrypted: Hello

Figure 3: Enhanced security mechanisms with uIP implementation.

implemented. We shall illustrate them briefly at different layers of the network layer, application layer and transport layer.

The code that was written that affected the µIP scheme is listed below

```
**
* Author: Celestine Iwendi
*/
#include "contiki.h"
#include "net/tcpip.h"
#include "node.h"
PROCESS(test_dos_process, "Test against DOS attack");
AUTOSTART_PROCESSES(&test_dos_process);
PROCESS_THREAD(test_dos_process, ev, data) {
PROCESS_BEGIN();
static µIP_ipaddr_t bsaddr;
static struct etimer timer;
µIP_ipaddr(&bsaddr, 172, 16, 1, 0);
while (1) {
etimer_set(&timer, CLOCK_SECOND);
PROCESS_WAIT_EVENT_UNTIL(etimer_expired(&timer));
if (ev == tcpip_event) {
struct message new;
new.type = JOIN_NETWORK_REQUEST;
µIP_send(&new, sizeof(new));
}
if (tcp_connect(&bsaddr, PREALLOC_PORT, NULL) == NULL) {
µIP_abort();
continue;
}
}
PROCESS_END();
}
```

Spoofing, replaying or altering clustering messages: One method to combat the threat is by authentication and antireplay protection. Preallocation of keys was the method that we used to prevent this attack as demonstrated by our scheme.

Hello flood attack: This form of attack that does not require the attacker to compromise encryption is difficult because nodes tend to broadcast hello messages in many protocols as a way to make their presence known to their neighbours. To combat this type of attack, a method known as pairwise authentication which allows the nodes to verify its bidirectional links before constructing the routes was used in our unique mesh scheme, where the shortest path randomly changes direction. Although a 100% protection cannot be guaranteed because of the limitation of Cooja. We had made sure that the network does not allow hackers but DoS is not hacking and every single system in the world is prone to DoS attacks, the only difference is, at what boundaries will the network crash. Our µIP network crashed after a long time of flooding due to limitation of the simulator as shown in Figure 3.

Homing attack: This attack that uses the traffic pattern to analyse and identify some targeted nodes with cluster heads or cryptographic functions was prevented by the use of AES which provides a header encryption mechanism.

The sybil attack: This attack that presents a malicious node as multiple identities to the network specially targeting the multipath routing was tricky. Our key validation for random key pre-distribution and position authentication prevented it.

Wormholes: This attack that depends on a node misrepresentation of its location, with a compromised node sending data between two valid nodes is a ruthless attack that is difficult and challenging to defend against in ad hoc and WSN. Our location based routing protocols had 50% potential stopping it but it is recommended to us 3D routing mechanism if we are to totally eradicate such attacks.

Synchronize flood attack: The attack that tends to exhaust memory resources of a network, with an attacker sending many connection establishment requests, forcing the network to allocate memory in order to maintain the state for each connection was not tested because of the limitation of Contiki OS. The primary defence would have been to use SYN (Synchronised) cookies, but this technique's computation makes them difficult in WSN as shown in Figure 4.

Finally, there are many other attacks that were not looked into with most of them similar and may require the same mechanism to prevent them intruding into the network. We can see that securing WSN is not an easy job from denial of service attack. Our scheme therefore can be trusted in mitigating these attacks with a minimal usage of memory capacity and low energy consumption but with higher reliability and better authentication technique. The key management technique also shows a great scalability advantage and maintains a significant impact on the performance metrics. The enhanced security method is therefore expected to demonstrate even better secured network with the integration of IP-enabled WSN nodes.

The overall energy consumption (Table 1 and Figure 5) can also be reduced significantly in the hardware to encourage the massive distribution of sensor nodes in areas that cannot easily be reached like under water sensing and hurricane monitoring as will be shown in the next section.

A closer look on the results shown on Figure 6 illustrate that the value of multihop transmission using Rime protocol is missing. That's because it takes so long that it causes request timeout each time the program is run. This is because of insufficiency of Rime protocol. The nodes were tested for both single-hop and multi-hop using the worst case scenario where each node sees only 2 neighbors and a message has to travel through each node in the network.

Conclusion

Securing and embedding a networking stack is no longer a big task that requires too much amount of resources. As we have shown, there exist a quite number of solutions that make TCP/IP suitable for WSN communications. The modification of µIP open source stack to accommodate the sensor nodes code using the key distribution method as an interface with the sensor nodes within an IP domain therefore provides an alternative of protocol implementation for the µIP-WSN

```
250      ID:6    Starting 'Receive key'
348      ID:4    Contiki 2.5 started. Node id is set to 4.
348      ID:4    Rime address: 4.0
348      ID:4    MAC nullmac RDC nullrdc NETWORK Rime
348      ID:4    IPv4 address: 172.16.4.0
348      ID:4    Starting 'Receive key'
382      ID:1    Contiki 2.5 started. Node id is set to 1.
382      ID:1    Rime address: 1.0
382      ID:1    MAC nullmac RDC nullrdc NETWORK Rime
382      ID:1    IPv4 address: 172.16.1.0
382      ID:1    Starting 'Send keys'
402      ID:7    Contiki 2.5 started. Node id is set to 7.
402      ID:7    Rime address: 7.0
402      ID:7    MAC nullmac RDC nullrdc NETWORK Rime
402      ID:7    IPv4 address: 172.16.7.0
402      ID:7    Starting 'Receive key'
556      ID:10   Contiki 2.5 started. Node id is set to 10.
556      ID:10   Rime address: 10.0
556      ID:10   MAC nullmac RDC nullrdc NETWORK Rime
556      ID:10   IPv4 address: 172.16.10.0
556      ID:10   Starting 'Receive key'
708      ID:9    Contiki 2.5 started. Node id is set to 9.
708      ID:9    Rime address: 9.0
708      ID:9    MAC nullmac RDC nullrdc NETWORK Rime
708      ID:9    IPv4 address: 172.16.9.0
708      ID:9    Starting 'Receive key'
715      ID:5    Contiki 2.5 started. Node id is set to 5.
715      ID:5    Rime address: 5.0
715      ID:5    MAC nullmac RDC nullrdc NETWORK Rime
715      ID:5    IPv4 address: 172.16.5.0
715      ID:5    Starting 'Receive key'
899      ID:3    Contiki 2.5 started. Node id is set to 3.
899      ID:3    Rime address: 3.0
899      ID:3    MAC nullmac RDC nullrdc NETWORK Rime
899      ID:3    IPv4 address: 172.16.3.0
899      ID:3    Starting 'Receive key'
27191650 ID:8    Preallocation is complete
45201344 ID:11   Preallocation is complete
52291744 ID:4    Preallocation is complete
64166744 ID:3    Preallocation is complete
72086345 ID:2    Preallocation is complete
96217260 ID:4    Sending decrypted = Hello, encrypted = V?a□□□^V?d□
————————————————8'
96217263 ID:3    Received encrypted: V?a□□□^V?d□————————————————8',
decrypted: Hello
111233844 ID:9   Preallocation is complete
128392759 ID:11  Sending decrypted = Hello, encrypted = ﾐｵ=□>□□□□□4
128392762 ID:3   Received encrypted: ﾐｵ=□>□□□□□4, decrypted: Hello
146647776 ID:2   Sending decrypted = Hello, encrypted = □
146647783 ID:3   Received encrypted: □, decrypted: Hello
174849259 ID:11  Sending decrypted = Hello, encrypted = ﾐｵ=□>□□□□□4
174849261 ID:3   Received encrypted: ﾐｵ=□>□□□□□4, decrypted: Hello
200337766 ID:9   Sending decrypted = Hello, encrypted = P3M){Z4□rw□ﾑ
200337771 ID:3   Received encrypted: P3M){Z4□rw□ﾑ, decrypted: Hello
```

Figure 4: uIP Crashed After Flooding.

States	µIP Protocol (pW)	Rime Protocol (pW)
Pre-allocation	0.1750	0.0900
Single-Hop Communication (Sender)	0.0200	0.0150
Single-Hop Communication (Receiver)	0.0150	0.0150
Multi-Hop Communication (Sender)	0.0300	0.0280
Multi-Hop Communication (Receiver)	0.0280	0.000

Table 1: Power Consumption Analysis of µIP and Rime using Enhanced scheme.

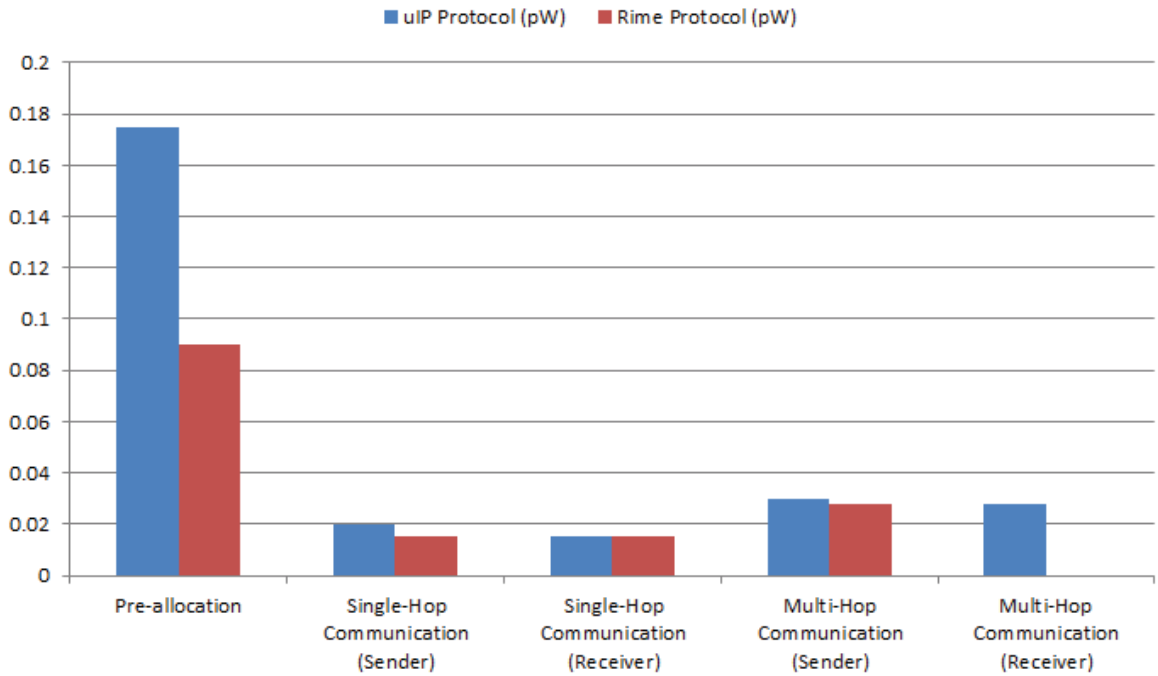

Figure 5: Power Consumption Analysis of uIP and Rime using Enhanced scheme.

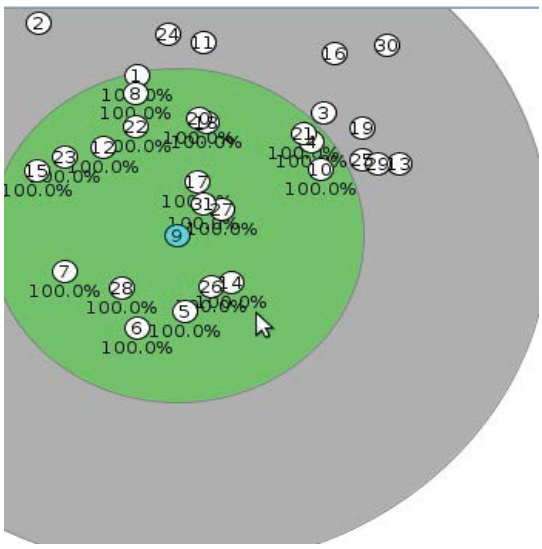

Figure 6: Irregularities at the network.

without creating problems of memory capacity and reducing the overhead cost. Each of the units is designed for both performance and confidentiality view point to allow the network to achieve connectivity, and multi-path balancing.

References

1. Iwendi CO, Allen AR (2011) CIA Security Management for Wireless Sensor Network Nodes. Proc. Annual Postgraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting, Liverpool, UK.

2. John JP, Katz-Bassett E, Krishnamurthy A, Anderson T, Venkataramani A (2008) Consensus Routing: The Internet as a Distributed System. Proc. NSDI Advance Computing Systems Association 351-364.

3. Dunkels A (2007) Rime-A Lightweight Layered Communication Stack for Sensor Networks. Proc. European Conference on Wireless Sensor Networks (EWSN), Delft, Netherlands.

4. Iwendi CO, Allen AR (2011) Wireless Sensor Network Nodes: Security and Deployment in the Niger-Delta Oil and Gas Sector. International Journal of Network Security & Its Applications 3: 68-79.

5. Iwendi CO, Allen AR (2012) Enhanced security technique for wireless sensor network nodes. Proc. IET Conference on Wireless Sensor Systems, London, UK.

6. Colitti W, Steenhaut K, De Caro N, Buta B, Dobrota V (2011) REST Enabled Wireless Sensor Networks for Seamless Integration with Web Applications. IEEE 8th International Conference on Mobile Adhoc and Sensor Systems, Valencia, Spain.

7. Granjal J, Monteiro E, Sa Silva J (2010) Enabling Network-Layer Security on IPv6 Wireless Sensor Networks. Proc. Global Telecommunications Conference (GLOBECOM 2010), Miami, Florida, USA.

8. Jelicic V (2011) Power Management in Wireless Sensor Networks with High-Consuming Sensors. Qualifying Doctoral Examination, University of Zagreb, Croatia.

9. Eghbali AN, Dehghan M (2007) Load-balancing using multi-path directed diffusion in wireless sensor networks. Proc. 3rd international conference on Mobile ad-hoc and sensor networks (MSN'07), Hongke Z, Stephan O, Jiannong C, David BJ, Springer-Verlag, Berlin, Heidelberg Germany.

10. Yarvis M, Kushalnagar N, Singh H, Rangarajan A, Liu Y, et al. (2005) Exploiting heterogeneity in sensor networks. Proc. IEEE INFOCOM, Miami, Florida, USA.

11. Young JK (2008) Untangling the Mesh: The Ins and Outs of Mesh Networking Technologies. Digi International Inc, USA.

12. Kvalbein A, Lysne O (2007) How can multi-topology routing be used for intradomain traffic engineering? Proc. SIGCOMM workshop on Internet network management, Kyoto, Japan.

13. Digi International (2009) Efficient Data Transfer over Cellular Networks. Digi International Inc, USA.

14. Das S, Yiakoumis Y, Parulkar G, McKeown N, Singh P, et al. (2011) Application-Aware Aggregation and Traffic Engineering in a Converged Packet-Circuit Network. National Fiber Optic Engineers Conference and Optical Fiber Communication Conference and Exposition, Los Angeles, USA.

15. Mayer K, Fritsche W (2006) IP-enabled wireless sensor networks and their integration into the internet. Proc. International Conference on Integrated Internet Ad Hoc and Sensor Networks, New York, USA.