Decentralized and Secure Protocol for Wireless Sensor Network to Provide Authentication and Integrity of the Distributed Data

P.Srivalli, VNRVJIET
M.Tech, CNIS (IT)

Abstract- Generally, a data discovered and distribution protocol inwireless sensor network is used for changing configuration parameters, commands related to distributing management to the sensor nodes in a network. All the present data discovered and distributed protocols experience from the two disadvantages. The initialdisadvantage is, base station is main for distributing the data to the sensor node i.e., these are based on the consolidate method. Last, all these protocols are not designed well regarding the securityof data issues. Here, we are using the first secure, reliable distributed data finding and distribution protocol named asDiDrip. DiDrip protocol uses multiple users with different access rights to spreadthe data items to the multiple sensor nodes in a network. AES algorithm is used to enhance the security of the project.

Keywords— *Finding of Data , Distribution, Wireless sensor networks,AES.*

I. INTRODUCTION

There are manyprotocols related to searching of data and distribution of data are proposed for Wireless sensor networks. Among them, the three protocols, DHV[1], DIP[2], Drip[3] are considered as the state-of-the-art protocols and they are also present in TinyOS. TinyOS is used by thousands of developersthroughout the world on different platforms for broad range of Wireless sensor networks. Drip is a network encoding protocol to improve efficiency, reliability and speed of distribution of sensor nodes. The other protocol DHV is a code stability preservation protocol for different Multi-hop wireless communication networks. DIP is density inference protocol which provides the information about the sensor nodes. Some Wireless sensor networks do not possessthe base stations.All the present searching of data and spreading protocols [4], [5] have been proposed for Wireless sensor networks.

Large number of sensor networks are implemented in various projects such asGeoss [6], NOPP [7], and ORION[8]. The three networks are operated by multiple owners and they are utilised by differentapproved third force users.Trickle [9] algorithm, a base algorithm for both DIP and Drip. Trickle can disseminate new data to the nodes very quickly. Among all these protocols, DiDrip protocol is based on decentralized approach. This protocol should satisfy the following requirements. They are:

1. Distributed
2. Supporting different user privileges
3. Authenticity and integrity of data items
4. User accountability
5. Node compromise tolerance
6. User collusion tolerance
7. DoS attacks resistance
8. Freshness
9. Low energy overhead
10. Scalability
11. Dynamic participation

In this project, a owner, group of users and a sensor network is included.

We also use 160 bit SHA-1, 160 bit ECCand 128 bit AES in implementation. We will discuss about them in later sections.

Now, we will look about the system architecture. It compares both existing and proposed approaches. By having a glance at the figure, we can easily understand the difference existing between them and their approaches. In existing only a base station and a sensor network is used and here only the base station takes responsibility for spreading the information to the various multiplesensor nodes. Here in proposed approach, an administrator of network, multiple users and a sensor network is included. Owner handles the multiple users and users construct the packets. These packets are disseminated to the sensor network without indulging the base station. So, it comes under secure and distributed approach whereas the other system is based on centralized approach.

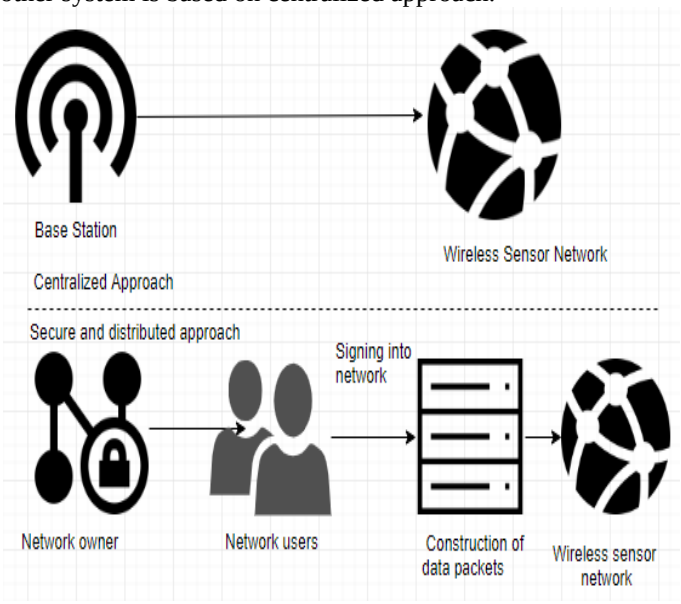
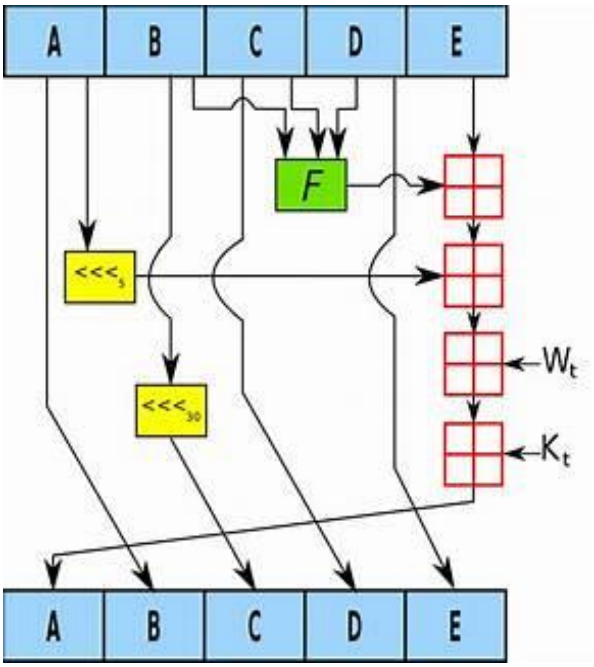


Fig.1: System Architecture

II. RELATED WORK

A. SHA-1:

In related work, we will discuss about SHA-1 and ECC. SHA abbreviation is secure hash algorithm. SHA-1 is one of a kind of cryptography hashing functions which considers dataas an input and gives us 160 bit hash value which is commonly said as message digest value- hexadecimal number which is 40 digits long. It is designed by USNS Agency. Block size is of 512 bits. It contains 80 rounds. Generally, hashing algorithms are used in many ways and some of them are for storing passwords, computer vision and databases. They also produce irreversible hashes and unique hashes. SHA-1 is also collision resistant i.e., two inputs cannot have same hash value. List of the protocols utilisingSHA-1 are Transport layer security(TLS), Secure socket layer(SSL), Pretty Good Privacy(PGP), Secure shell(SSH), Internet protocol security(IPSec) etc. This algorithm consists of some couple of steps where the text or numbers are converted into 40 bit hash code. For example,the SHA-1 code for 'test' is a94a8fe5ccb19ba61c4c0878d391e987982fbbd3. The same code repeats for the test and another one cannot have this same hash value.



. Fig.2: SHA-1

In this context, SHA-1 is used for data integrity and the values disseminated by the nodes are verified with SHA-1 signature. In existing system, only a base station and a sensor network is present. If any problem occurs in the base station, such as if climatic conditions are not good then the signals cannot be reached by the sensor network. In this way the connection is not established and data cannot be transformed. To eradicate this problem ,we are using secure and distributed approach.

B. ECC:

160 bit ECC is used here. ECC abbreviation is Elliptic curve cryptography. In ECC we have different bits and they are 112,160,224,256,384 and 512. ECC comes under Asymmetric cryptography algorithm. First, we will discuss about Cryptography. Cryptography is transforming the messages from one to another to make them secure and immune to attack. It holdsboth encryption and decryption. Encryption is a process of transformation of plain text into cipher text (which is not understandable by a user). Decryption is the opposite of this process. In Cryptography, we have two categories Symmetriccryptography algorithmsand Asymmetric cryptography algorithms. Symmetriccryptography uses similar key for both encryption and decryption. Asymmetric cryptography uses two keys i.e., public and private keys are used for both encryption and decryption. A key is a value represented in terms of variables that is implied using an algorithm to a block of a text or a string to produce encrypted text or to decrypt the encrypted text. ECC comes under Asymmetric algorithm. Among all the algorithms the key generation in ECC is very fast. It also uses small size keys, cipher texts and signatures. It also uses less memory space when compared to another algorithm. The general curve equation for ECC is $y^2 = x^3 + ax + b$.It also contains two fields, one is prime field and another one is infinite field.

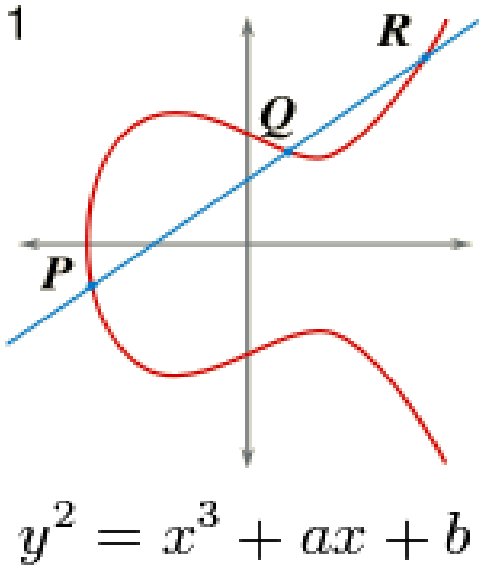


Fig.3: General Elliptic Curve

Randomly a point is selected in the curve and by using the private key and the generated point, we are deriving the public key. Point addition, doubling and multiplication is also performed here.

In the existing system, the experiments are conducted on two motes and they are MicaZ, TelosB. They have used Ubuntu 11.04 environment.

MicaZ and TelosB motes are the building blocks of Wireless sensor networks. They are cheap in cost and small in size. Generally, a sensor consists of microcontroller, transceiver, external memory and power source. C programming and nesC is used. They are implemented with both Data hash chain method and Merkle hash tree. Results shown with thesecomparisons between these methods. DiDrip1 is the

enhancement of DiDrip protocol. The main advantage of this protocol is, attack resistance and it is based on the distributed approach. DiDrip2 includes the implementation of DiDrip1 with message specific puzzle approach. First, we will see about the DiDrip protocol.

C. DiDrip:

DiDrip holds four stages. They are:
1) System Initialization Phase
2) User Joining Phase
3) Packet Pre-Processing Phase
4) Packet Verification Phase

D. System Initialization Phase:

In this Initialization Phase, first 160 bit ECC is setup. Here, 160 bit ECC is used for generation of public and private keys.x is private key, y a public key and some public parameters values {y,Q,p,q,h(.)} are loaded. Q is a point on curve E. Two big prime numbers are used- p,q and h is a hash function (SHA-1). P, q values is 160 bits i.e., p is 80 bits and q is 80 bits. Here, an Elliptic curve E is selected over GF(p) which is a finite field. Private key is selected, x ϵ GF(q) and a public key is computed y = xQ.

E. User Joining Phase:

Each user has its own identity Uj andobtain dissemination privilege(Pri_j). Here the length of Uj is 2 bytes(16 bits) and it can support 65,536 users. User sends a 3 data fields<UIDj,Pri_j,PKj> to the network administrator.After receiving this message, a certificate is generated $Cert_j$ and a certificate composes of {UIDj, PKj, Pri_j, SIG_x{h(UIDj||PKj||Pri_j)}, value of Pri_j is 6 bytes and certificate is of 88 bytes(704 bits).

F. Packet Pre-Processing Phase:

Users joinsinto the network and they need to spread some information; d_i={key, version, data} i value varies from 1,2,3,….n. For formation of these packets, Merkle hash tree is used.

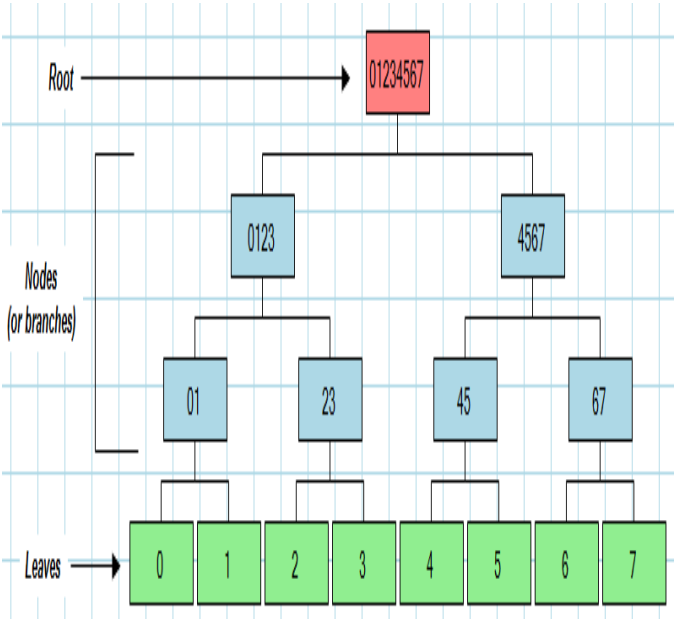


Fig4: Merklehash Tree

User builts the merkle hash tree from n number of information items. All theseinformation are treated as leaf nodes of a hash tree. A bunch of new nodes are formed at top layer of the tree. Each node is formed by sequence of two child nodes. This process continues till the parent node is established. Before dissemination of n data items, user Uj designates the parent(main) node with their own SKjand a packet is formed which comprises of Certj, Hroot and $SIG_{SKj}\{H_{root}\}$. Here the certificate consists of user identity and Pri_j.

G. Packet Verification Phase:

Last phase of DiDrip protocol. Sensor nodes receives the packet from user Uj, it then checks the packets key field and legality of dissemination privilege Pri_j. Though the result is beneficial, node Sj uses the y value of administrator to run an ECDSA verification operation to check thegenerated certificate. For suppose it is correct, then it checks the authenticityof the certificate. For suppose, authentication is failed, then the packet is discarded.
This is the detailed description of the four phases of DiDrip protocol.

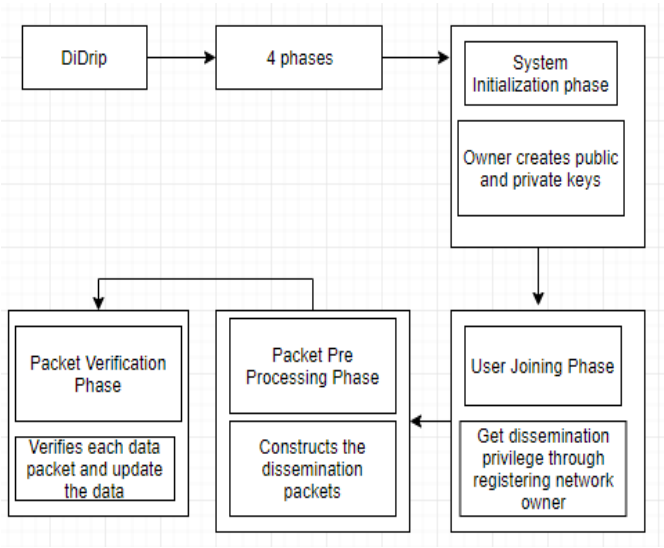


Fig.5: DiDrip protocol

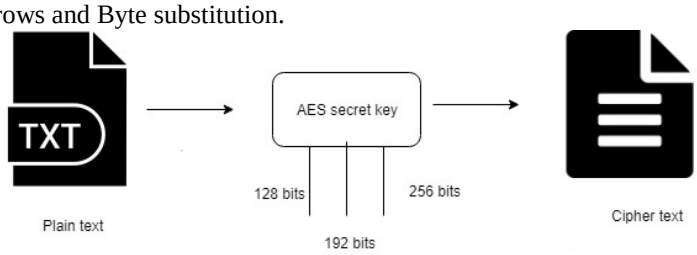
In the previous section, we have discussed about the motes. We have used two motes in our related work i.e., Micaz and Telosbmotes . Each mote has its own configuration values and parameters. They are compatible on TinyOS 2.x version. Network owner and user programs are written in C programming language not in java using OpenSSL [10].OpenSSL is a full featured and robust toolkit for the both transport layer security [TLS] and secure socket layer [SSL] protocols. All the experiments were repeated 1000 times for accurate results.

In DiDrip1, message specific approach is not implemented. Here in this context, a bunch of information is received by the sensor nodes,then authentication of messagesis checked before distributingthe data to the other side nodes. In contrast to DiDrip1, DiDrip2 is implemented with message specific puzzle approach. According to this context, a sensor nodesonly checks the solutions of puzzles before broadcasting it to other ones. Actually, we have two characteristics of these puzzles. One of these are, the puzzles are not easy to get solved but their solutions can be verified very easily and simply. Second one is, the puzzle has a certain limitations of time for solving. The advantage of this approach is, it reduce the dissemination delay i.e., there will be no delay in dissemination of data packets to the nodes. At last, merkle hash tree method is efficient when compared to data hash chain method based on the comparative results. In this implementation results, the CPU execution times were derived during the important operations of DiDrip. This time is different for different phases of protocol. Execution times of SHA-1 hash function for both the motes vary in their values.

III. CONTRIBUTION

A. AES:

In the existing work, the security is lacking. So, to enhance the security we have implemented AES algorithm. AES comes under the category of Symmetric cryptography algorithms. It uses only one key i.e., secret key for both the encryption and decryption. Here, in this context, 128 bit AES is used. AES consists of another bits – 192,256. 128 bit AES consists of 10 rounds. Most common key used is 128 bits. We have implemented AES because, it is very fast and it can easily be implemented in Java. It is of six times faster than of triple DES(Data Encryption Standard). Working of AES is based on the principle of 'Substitution – Permutation network'. AES executes its operations on bytes rather than bits. 128 bits of data are treated as 16 bytes (1 byte = 8 bits). Generally, the encryption process consists of four phases - Byte Substitution, Shift rows, Mix columns and Add round key process. Decryption process is the reverse process of encryption. Decryption possesses Add round key, Mix columns, Shift rows and Byte substitution.



B. Proposed System:

- In proposed system, for front end we have used Swings and AWT (Abstract Windowing Toolkit).
- AWT is platform dependent and it contains heavyweightcomponents. It also uses more memory space.
- Swings are reverse of AWT. It is platform independent, it uses less memory space and it is faster when we compare it to the AWT. The main characteristic of Swing is, it provides a flexible user interface and it designs the user interface based on the concept of MVC(Model View Controller).
- We have used Java programming language for coding section and operating system is above Windows XP version. For back end part, MySQL 6.0 version is used for storing the data.
- We have used eclipse, an integrated development environment, which is widely used for developing java applications. It is free and an open source software. First, we have created a java project and drip package is created.
- We have created modules for data base connection, main, owner login, user login, simulation screen, ECC, SHA-1,AES and graphs.
- Each module consists of code and all the modules are interlinked. Each module has its own functionality.
- First, an owner logins into the network with 'username' and 'password' is admin. If an owner types wrong username or password, then it will account for an invalid login.
- After immediate login of owner, register the users with certain privileges. After adding the users, a message 'user details added successfully' is displayed on the screen. Here, privileges are nothing but, a user can access certain limit of nodes only. Owner can register multiple users.

- In background, ECC algorithm runs and generates the public and private keys. These keys are issued to the users and a simulation screen is displayed with 50 sensor nodes with numbering. Owner logout from the network.
- Now, the user login into the network and join the network with certain privileges. Here, we have used 50 sensor nodes and suppose one user has right to discover the data and disseminate to the 0-5 nodes of a network.
- Multiple users have different privileges. After user login, user joins into the network and issue commands to the nodes.
- For example, the commands are only integers and value given is of user choice. It supposed to be in a format of 10-50 where, 10 is the minimum value and 50 is the maximum value.
- These numbers are of our choice and there is no limit for it i.e., unlimited. After issuing the commands, AES algorithm with its key is used to encrypt the values such that the values are secured.
- Values are generated in range of minimum value and maximum value from respective sensor nodes of a particular user.
- These generated values are verified with SHA-1 and a unique 40 bit hash code isgenerated. It is different for different values.
- Suppose, if we have used 5 nodes and 10 commands then the 50 simulations took place.
- Charts are generated for each user with different privileges and it composes of execution time and memory usage.
- Execution time is generated in terms of Nano seconds(10^{-9}) and memory usage is generated in terms of kilo bytes. At last, user logout from the network.
- In this way, multiple users with multiple privileges had discovered their data and generated it from the couple of nodes securely without relying on base station.

IV. SECURITY ANALYSIS

A. DoS Attack:

Denial of Service attack (DoS) is one of a attack in cyber security. The main intention of this attack is, attacker takes over the control of the resource or a machine, making unavailable to the authorised users, indirectly the connection is disrupted or disconnected. In existing system, the DoS attack took place because of no proper security.

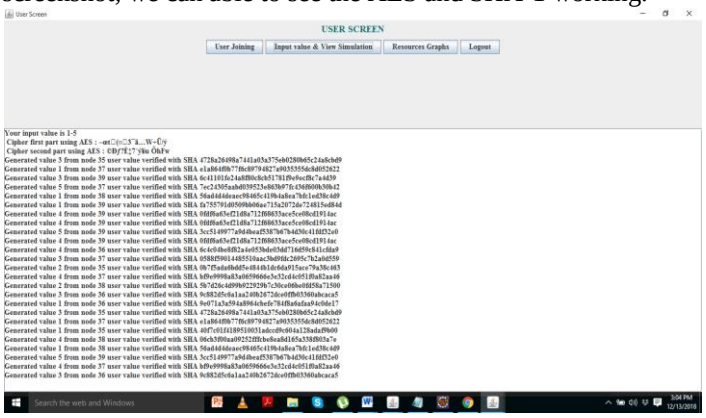
In proposed system, we have improved the security mechanisms and some of the values are encrypted and some values are produced into hash code which is irreversible. In this way, the intruder cannot enter into the network to steal or to alter the data.

B. User accountability:

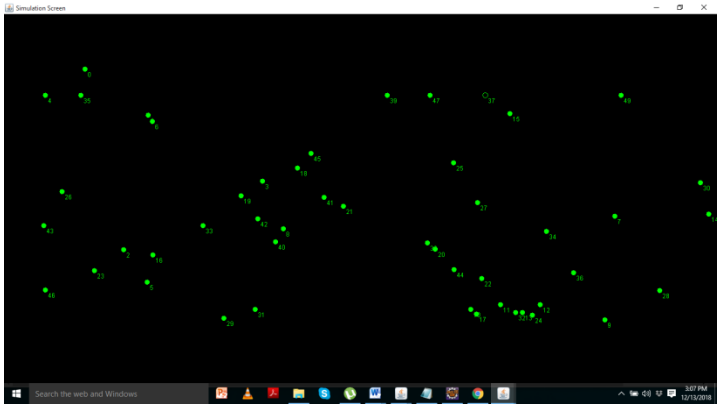
Identification of users and their dissemination of data activities are known to the sensor nodes. As this is the case, the nodes can report the issues to the network admin. Only the network administrator has the responsibility of modifying the user activities.

V. RESULTS

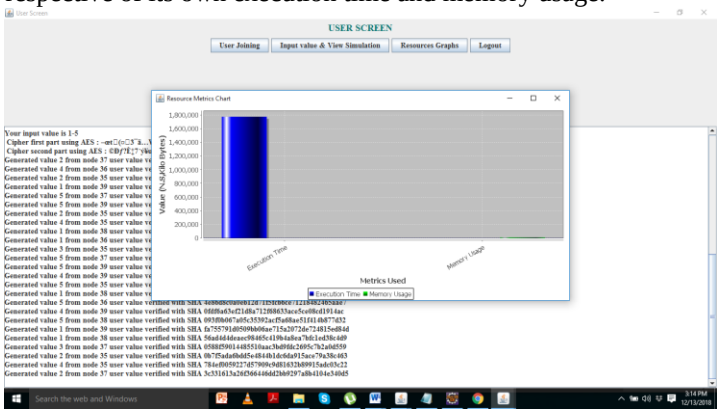
Visual representation is necessary for coding projects. We get a good idea about the project and we cannot forget easily. We have created owner screen, user screen and simulation screen. Each screen has its own buttons and each button has its own functionality. However the user interface plays a vital role in designing the projects. By have a glance at screenshots, we get a good idea about the project.When a user login into the network with its unique username and password, want to discover the data and issue some commands. This command consists of a minimum value and a maximum value. In this screenshot, we can able to see the AES and SHA-1 working.



When the values are generating from sensor nodes, the node blinks or repaint such that it is reading the values. It is clearly seen in the below simulation screen.



After this whole process, we generate a graph for each user respective of its own execution time and memory usage.



VI. CONCLUSION

In previous existing system, the security for data is improper. To be safe from intruders, we have used AES algorithm to protect our sensible information against the attackers. We had put our whole concentration on AES and SHA-1 algorithms and less concentrated on ECC. We have clearly implemented the DiDrip protocol which is based on the distributed approach and transformation is performed securely without having a base station. We have seen many security issues in discovering the data items and distribution of data items to end users in Wireless sensor networks, which is addressed in previous researches. So, we have addressed these adversaries in our proposed system.

VII. REFERENCES

[1]. T. Dang, N. Bulusu, W. Feng, and S. Park, "DHV: A code consistency maintenance protocol for multi-hop wireless sensor networks," in Proc. 6th Eur. Conf. Wireless Sensor Netw., 2009, pp. 327–342.

[2]. K. Lin and P. Levis, "Data discovery and dissemination with DIP," in Proc. ACM/IEEE Int. Conf. Inf. Process. Sensor Netw., 2008, pp. 433–444.

[3]. G. Tolle and D. Culler, "Design of an application-cooperative management system for wireless sensor networks," in Proc. Eur. Conf. Wireless Sensor Netw., 2005, pp. 121–132.

[4]. M. Ceriotti, G. P. Picco, A. L. Murphy, S. Guna, M. Corra, M. Pozzi, D. Zonta, and P. Zanon, "Monitoring heritage buildings with wireless sensor networks: The Torre Aquila deployment," in Proc. IEEE Int. Conf. Inf. Process. Sensor Netw., 2009, pp. 277–288.

[5]. D. He, S. Chan, S. Tang, and M. Guizani, "Secure data discovery and dissemination based on hash tree for wireless sensor networks," IEEE Trans. Wireless Commun., vol. 12, no. 9, pp. 4638– 4646, Sep. 2013.

[6]. Geoss. [Online]. Available: http://www.epa.gov/geoss/

[7]. NOPP. [Online]. Available: http://www.nopp.org/

[8]. ORION. [Online]. Available: http://www.joiscience.org/ocean observing/advisors

[9]. P. Levis, N. Patel, D. Culler, and S. Shenker, "Trickle: A self-regu-lating algorithm for code maintenance and propagation in wire-less sensor networks," in Proc. 1st Conf. Symp. Netw. Syst. Design Implementation, 2004, pp. 15–28.

[10]. OpenSSL. [Online]. Available: http://www.openssl.org