International Journal of Engineering Research ISSN:2319-6890)(online),2347-5013(print)
Volume No.5 Issue: Special 4, pp: 790-991 20 May 2016

Integrity of Data Discovery and Dissemination to Improves the Quality of Service (QoS) in WSNs

Ashwini[1], Dr. Chandrakant Naikodi[2], Dr. L. Suresh[3]
[1]MTech (CSE)-4th Semester,
Department of computer science and Engineering
Cambridge Institute of Technology, Bangalore, India
[2]Visiting Professor, Dept. of Computer Science and Engineering,
Cambridge Institute of Technology, Bangalore, India
[3]Professor and Principal
Cambridge Institute of Technology, Bangalore, India

ABSTRACT

Data dissemination is the process by which queries of data are routed in the sensor network. The data collection by sensor nodes has to be communicated to the base station or to any other node interested in the data. Data discovery and dissemination protocol for WSNs is responsible for updating configuration parameters of and distributing management commands, to, the sensor nodes. All existing data discovery and dissemination protocol suffer from two drawbacks.

First, they are based on the centralized approach; in centralized approach only the base station can distribute data items, such an approach is not suitable for emergent multi-owner--multi-user wireless sensor networks. Second, those protocols were not designed with security in mind and hence adversaries can easily launch attacks to harm the network. This paper proposes the first secure and distributed data discovery and dissemination protocol called DiDrip. It allows the network owners to authorize multiple network users with different privileges to simultaneously and directly disseminate data items in to the sensor nodes based on the design objective we propose DiDrip. This is the first distributed data discovery and dissemination protocol, which allows network owners and authorized users to disseminate data items into WSNs without relying on the base station. Moreover our extensive analysis demonstrates the DiDrip satisfies the security requirements of the protocol. Here implements the multi hop data transmission in the network while transmitting data prioritization given to the data. There are three types of data packets low, medium and high by using dynamic scheduling algorithm and improves the quality of service.

1. INTRODUCTION

A wireless sensor network sometimes called a wireless sensor and an actuator network [1], that are spatially distributed autonomous sensors to monitor physical or environmental conditions [2], such as a temperature, sound, pressure, etc. and cooperatively pass their data through the network to a main location. The Wireless Sensor Networks is built "nodes"- from a few to several hundreds or even thousands where each node is connected to one sensor. Each sensors network node has several parts; a radio transceiver with an external antenna, a microcontroller, an electronic circuit for interfacing with the sensors and an energy source, usually a battery or an embedded form of energy harvesting. Here the Sensor nodes might vary in size from that of a shoebox down to the size of a grain of dust, although functioning "motes" of genuine microscopic dimensions have yet to be created. The cost of sensor nodes is similarly variable ranging from a few to hundreds of dollars, developing on the complexity of the individual sensor nodes, size and cost contains on sensor nodes, result in corresponding constraints on resources such as energy, memory, computational speed and communications bandwidth. As we told in above wireless sensor network is made of sensor nodes used for monitoring and analysis purpose as shown in Fig 1. These sensor nodes pass the information that they collect to a prime location called base station. In most systems, a WSN communicates with a LAN or WAN through a gateway like medium. The gateway is actually a bridge between the WSN and various other sensor networks. This allows the data to be stored by devices and which can be taken up for processing later.

After a wireless sensor network is deployed there is usually a need to update buggy or old small programs or parameters stored in the sensor nodes. This can be achieved by the so called data discovery and dissemination protocols, which facilitates a source to inject small programs, commands, queries, and configuration parameters to sensor nodes. Note that it is different from the code dissemination protocol [3] [4]. Several data discovery and dissemination protocols [5] [6] [7] [8] have proposed for Wireless Sensor Networks. Among them, DHV [5], DIP [7] and Drip [6] are regarded as the state of the art protocols and have been

included in the Tiny OS distributions. All proposed protocols assume that the operating environment of the WSN is trustworthy and has no adversary.

Fig.1. Example of WSNs

However, in reality, adversaries exist and improve threats to the normal operation of the wireless sensor network [9] [10]. This issue has only been addressed recently by [9] which identifies the security vulnerabilities of Drip and proposes an efficient solution.

In this paper mainly consists of two approaches first one is centralized and the second ne is distributed in centralized approach data items can only be disseminated by the base station. The disadvantage of centralized approach is there may be chances of suffering the single point of failure as dissemination is impossible when the base station is not works properly or when the connection between the base station and node is broken.

Externally the centralized approach is inefficacious, equanimity, and vulnerable to security attacks that can be launched anywhere along the communication path [4]. Even worse case some WSNs do not have any base station. For example, the WSNs monitoring human trafficking in a country is border or a WSNs deployed in a remote area to monitor illegal or forbidden ablate cultivation, a base station can becomes an attractive target to be attacked. In such a network, data circulation is better to be carried out by the owner or authorized network users in a distributed manner. Furthermore, a distributed data discovery and dissemination named DiDrip is very relevant in wireless sensor networks. In shared sensor networks where sensing or communication infrastructures from multiple owners will be shared by the applications from multiple users. For example there is a large scale sensor networks built in recent project such as GEOSS [11], NOPP [12], and ORION [13]. These networks are owned by multiple owners and used by various authorized third party users.

Motivations by the above observations, this paper has the following main contributions:

- The need of distributed data discovery and dissemination protocols is not completely new, but previous work did not address this nee. We study the functional requirements of such protocols, and set their design objectives. Also, we identify these security vulnerabilities in existing data discovery and dissemination protocols.
- Based on the design objectives, we propose DiDrip. It is the first secure and distributed data discovery and dissemination protocol, which allows network owners and authorized users to disseminate data items into the WSNs without relying on the base station. More ever our extensive analysis demonstrates that DiDrip satisfies the security requirements of the protocols of its kind. In particular, we apply the provable technique to formally prove the authenticity and integrity of the disseminated data items in DiDrip.
- Also demonstrate the efficiency of DiDrip in practice by implementing it in an experimental WSN with resource limited sensor node. This is also the first implementation of a secure and distributed data discovery and dissemination protocol.

II. RELATED WORK

2. Threat Models and DiDrip

Mainly there are three types of models as shown below

2.1 Network Model

Figure 2 shows a general Wireless Sensor Networks include a large number of sensor nodes. It is administrated by the owner and accessible by so many users. The sensor nodes are usually to be a resource constrained with respect to memory space computation capability, bandwidth and power supply. Thus a sensor node can only perform a limited number of public key cryptographic operations during the period of its battery. The network users use some mobile devices to disseminate data items into the network. The network owner is responsible for generating keying material.

ISSN:2319-6890)(online),2347-5013(print)
20 May 2016

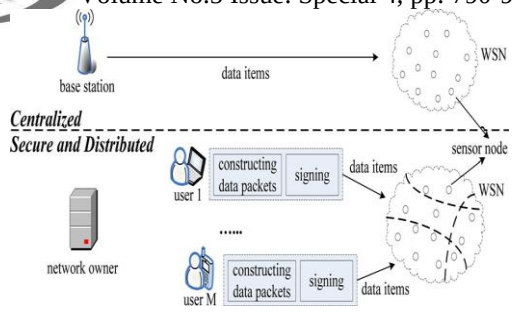


Fig.2. Data discovery and dissemination approachesof centralized and distributed

2.2. Trust Model

The network owner delegates that code dissemination privilege to the network users who are willing to register. We assume the special modules (e.g. authentication module for each new program image proposed in this paper, the user access log module) However, the network owner may for various reasons, impersonate network users to disseminate data items. Each sensor nodes cannot be overwritten by anyone except network owner.

2.3. Threat Model

Assume that adversary can launch both outsider and insider attacks. In outsider attack adversary does not control any valid nodes in WSN. Instead, it would eavesdrop or listen in for sensitive information, inject forged messages, launch replay attack, wormhole attacks, DoS attack and impersonate valid sensor nodes. The communication channel may also be jammed by the adversary, but this can only last for a certain period of time after which the adversary will be detected and removed.

By compromising either network users or sensor nodes, the adversary can launch insider attacks to the network. The compromised entities are regarded as insiders because they are members of the network until they are identified. The adversary controls these entities to attack the network in arbitrary ways. For instance, they could be instructed to disseminate false or harmful data, launch attacks such as Sybil attack or Dos attack, and be non-cooperative with other nodes.

III. PROBLEM STATEMENT

3. Data discovery and dissemination of security vulnerability

3.1 Review of existing Protocols

The underlying algorithm of both DIP and DRIP is Trickle. This requires each node to periodically broadcast a summary of its stored data. When node has received an older summary, it sent an updates to the source. Once all nodes have unchanging data then the broadcast interval is increased exponentially to save energy. In other words, Trickle can disseminate newly injected data very quickly. Among existing protocols the Drip is the simplest one and it runs an independent instance of Trickle for each data item.

In practice each data item is identified by unique key and its freshness is indicated by a version number. For example Drip, DIP, and DHV, each data item is represented by a 3-Tuples <key, version, data>, where key is used to identify a uniquely identify a data item, version indicates the freshness of the data item(the larger the version, the fresher the data), and data iindicates the actual disseminated data (for example, command, query or parameter).

3.2 security vulnerabilities

An adversary can first place some intruder nodes in the network and then use them to alter the data being disseminated. This may result in some important parameters being erased or the entire network is being rebooted with wrong data. For example, consider a new data item (key, version, data) being disseminated. When a raider receives this new data item, then it can broadcast a malicious data item <key, version*, data*> where version*> version. If data* is set to 0, the parameter identified by key will be erased from all sensor nodes. Alternatively, if data* is different from data, all sensor nodes will update the parameters according to this forged data item. Note that the above attacks can also be launched if an adversary compromises some nodes and has access to their key materials.

In addition, since nodes executing trickle are required to forward all new data items that they receive and an adversary can launch denial of service (DoS) attack to the sensor nodes by injecting a large amount of bogus data items. As result, the processing and energy resource of nodes are expended to process and forward these bogus data items, rather than on the intended functions. Any data discovery and dissemination protocol based on Trickle or its variants is vulnerable to such a DoS attack.

IV. PROPOSED SYSTEM

4. DiDrip

DiDrip is the first secure and distributed approach of data discovery and dissemination of data items to the sensor nodes. The application by multiple users share the communication infrastructure and sensing infrastructure of the multiple owners and the different users. There are four phases in DiDrip, system initialization phase, user joining phase, packet pre-processing phase and packet verification phase.

4.1 System Initialization Phase

System initialization phase is compared with the traditional approaches. Elliptic curve cryptography (ECC) is better approach to public-key cryptography in terms of key size,

doi : 10.17950/ijer/v5i4/010

computational efficiency. However, while ECC is feasible on resource-limited sensor mote, heavily involving ECC-based authentication is still not in practical. Didrip combines both ECC public key algorithm and merkle hash tree algorithm to avoid frequent public key operations and achieve strong robustness against various malicious attacks. Before the network deployment the private key x and some of the public parameters {y, Q, P, q, h (.)} are derived by the network owner and are preloaded in each sensor nodes.

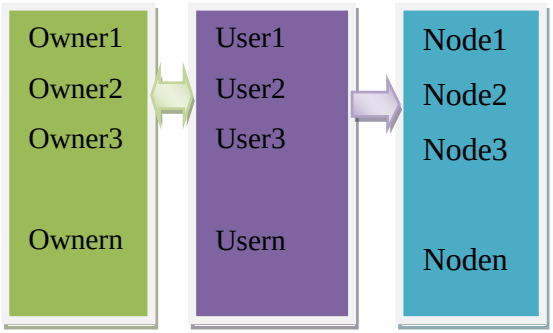
4.2 User Joining Phase

When the user uj needs to obtain privilege level it sent 3 tuples < UIDj, Prij, PKj > to the owner and the owner will compute,

Certj={UIDj,PKj,Prij,SIGx{h(UIDj||Prij)}

4.3 Packet Pre-processing Phase

When the user needs to disseminate data items d_i = {key_i, $version_i$, $data_i$} is must construct the data packets using two methods, i.e. data hash chain method and merkle hash method. Any of these methods can be used based on the WSNs characteristics.



Network Owner Authorized user Sensor Nodes

Fig3. Architecture

4.4 Packet Verification Phase

The sensor nodes on receiving a packet P_i it first checks the key field whether the packet is an advertisement packet or the data packet.

For the advertisement packet validation of the certificate and authentication of the signature is carried out. If yes, for the data hash chain method, the node Sj stores < UIDj, H1> otherwise, node Sj simply discards the packet. It is the data packet then the sensor nodes checks for the authenticity and integrity of the packet by comparing hash value of p_i with h_i, stores in P_i, when the result is positive; otherwise P_i is discarded.

Remark: To prevent the network owner from impersonating users, system initialization and issues of user certificates can be carried out by the certificate authority of a PK1 rather than the network owner.

V. LITERATURE SURVEY

After a wireless sensor network is deployed, there is usually a need to update buggy or old small programs or parameters stored in the sensor nodes. This can be achieved by so called data discovery and dissemination protocol, this can facilitates a source to infuse small programs, commands, queries, and contour parameters to sensor nodes.

D. He, C. Chen, S. Chan and J. Bu Proposed "Dicode: denial-of-service- resistant and distributed code dissemination in wireless sensor networks," which circulate large binaries to reprogram the whole network of sensors. For example, efficiently disseminating a binary file tens of kilobytes requires a code dissemination protocol while disseminating several two- byte contour parameters require data discovery and disseminate protocol. Considering the sensor nodes could be distributed in a comfortless environment, remotely disseminating such small data to the sensor nodes through the wireless channel is a more preferred and practical approach than manual mediation.

Disadvantage: through the wireless channel Remotely disseminating such small data to the sensor nodes is a more preferred.

T. Dang, N, Bulusu, W. Feng and S. Park proposed "DHV: A code consistency maintenance protocol for multi-hop wireless sensor network," an efficient code consistency maintenance protocol to certify that every node in a network will ultimatly have the consistent code. DHV is based on the simple observation, if two code versions are disperate, their corresponding version numbers often differ in only a few least significant bits of their binary representation. DHV allow nodes to carefully select and transmit only necessary bits level information to notice a newer code version in the network. DHV can recognize and identify version differences in O (1) messages and latency compared to the algorithmic scale of current protocols. The proposed protocol assumes that the operating environment of the WSN is trustworthy and has no adversary.

Disadvantages: Adversaries exist in reality and impose threats to the normal operation of WSNs.

They are based on the centralized approach.

K. Lin and P. Levis Proposed "Data discovery and dissemination DIP," Prior approaches, such as Trickle or SPIN, have atop that scale linearly with the number of data items. For T items, DIP can identify new items with O(log(T)) packets while maintaining a O(1) detection latency. To achieve this performance in a wide spectrum of network conformation, DIP uses a hybrid approach of randomized scanning and tree-based directed search. By dynamically selecting two algorithms to use, DIP outperforms in terms of transmissions and speed.

Disadvantages: Adversaries can easily launch attacks to harm the network. They are based on the centralized approach.

D. He, S. Chan, S. Tang, and M. Guizani Proposed "Secure data discovery and dissemination based on hash tree for wireless sensor networks," identifies the security vulnerabilities in data discovery and dissemination when used in WSNs identifies the security vulnerabilities in data discovery and dissemination when used in Wireless Sensor Networks. Such vulnerabilities allow an adversary to update a network with undesirable values, vulnerabilities, this paper presents the design, implementation, and evaluation of a secure, lightweight, and Dos-resistant data discovery and dissemination protocol named SeDrip for Wireless Sensor Networks. Our protocol takes into analysis the confined capability resources of sensor nodes, packet loss and out-of-sequence packet conveyance. Also, it can provide instantaneous authentication without packet buffering delay, and tolerate compromise.

Disadvantage: This paper suffers from the single point of failure as dissemination is impossible when the base station is not functioning or when the correction between the base station and a node is broken.

The centralized approach is inefficient, non-scalable, and vulnerable too security attacks that can be launched anywhere along the communication path.

VI. WORKING

6. Design and implementation

We evaluate DiDrip by implementing all components on an experimental test-bed. Also we can choose Drip for performance comparison.

6.1 Implementation and experimental setup

According to this paper we have written programs that will event functions of network owner, user, and sensor node. The network owner and user side programs are C programs using open SSL [20] and running on laptop PCS under ubuntu 11.04 with 2GB RAM environment. For sensor nodes also we written a program in nes C and run on resource limited motes (Mica Z and Telos B). The mica Z mote has an 8-bit 8-MHzAtmel microcontroller with 4kB RAM, 128-KB ROM, 512 KB of flash memory. Also Telos B mote has an 8-MHz CPU, 10-KB RAM, 48-KB ROM, 1MB of flash memory, and an 802.15.4/ZigBee radio. Our motes run Tiny OS 2.x. Additionally, SHA-1 is used, and the key sizes of ECC are set to 128 bits, 160 and 192 bits, respectively, Throughout this paper, all experiments on PCs (respectively sensor nodes) were repeated 1000.000 times is apply also for 1000 times, for each measurement in order to obtain accurate average results.

To implement DiDrip with data hash chain method (respectively, the merckle hash tree method), the following functionalities are added hash chain (respectively merkle hash tree) of a round of dissemination data, generation of the signature packet. For obtaining version number of each data items, the Disseminator C and Disseminator P modules in the Drip nesC library has been modified to provide an interface called Disseminator version. Moreover to proposed hash tree method is implemented without and with using the message specific puzzle approach presented in appendix.

In DiDrip1, when a node receives a signature data packet with a new version number, it authenticates the packet before broadcasting it to its next-hop neighbors and on the other side in DiDrip2, a node only checks the puzzle solution in the packet before broadcasting the packets. Based on the design of DiDrip, we implement the verification function for signature and data packets based on the ECDSA verify function and SHA-1 hash function of TinyECC2.0 library [21] and add them to the Drip nesC library. Also, in our experiment when a network user (i.e. laptop computer) disseminates data items, it first sends them to the serial port of specific sensor nodes in the network which is referred to as repeater. Then the repeater carries out the dissemination on behalf of the use using DiDrip.

The following matrices are used to evaluate DiDrip; memory overhead, execution time of cryptographic operations and propagate delay and energy overhead.

	ECC-128	ECC-160	ECC-192
Time (MicaZ)(ms)	2310	2436	3754
Time (TelosB) (ms)	3994	3955	5775

Table:1 Running Time for ECC signature Verification

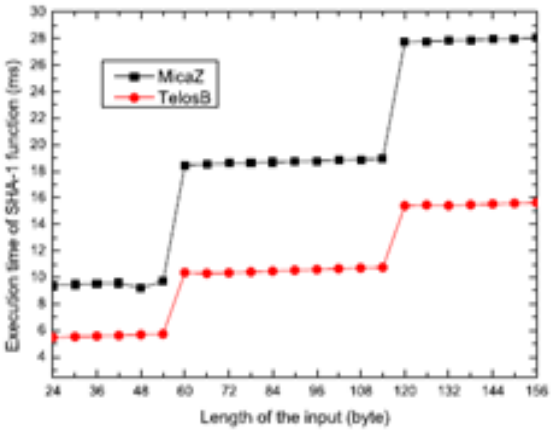


Fig: 3 The execution time of SHA-1hash function on MicaZ and TelosB motes

VII. Security and performance analysis

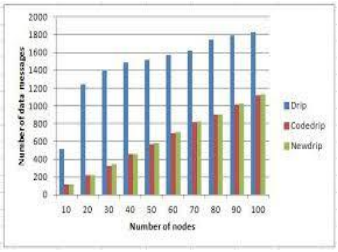
First we perform and analyze the security by this proposed protocol

- Resistance to pollution attack- attackers can't pollute the network with bogus data since data transfer done is always verified by cryptographic technique.

- Light-weight- Only simple at mathematical operations and encryptions techniques are used. Hence no much resource usage in nodes.

- Resistance to Denial-of-Service attack- Immediate authentication of packets is done at each destination, so it will discard fake packets and only valid packets pass through.

- Real time key generation- means do not fix an opening value for keys in nodes; which are calculated at time of data transfer only.



VIII. Conclusion and future work

In this paper we have identified the security vulnerabilities in data discovery and dissemination when used in Wireless sensor networks which have been not addressed in previous research. Also some of data discovery and dissemination protocols have been proposed, but none of those approaches support distributed operation. Therefore, in this paper, a secure and distributed data discovery and dissemination protocols named DiDrip has been proposed. Besides analyzing the security of DiDrip, this paper has also reported the evaluation of DiDrip in an experimental network of resource-limited sensor nodes, which shows that DiDrip is feasible in practice. We have also given a formal proof of the authenticity and integrity of the disseminated data items in DiDrip. Also due to the open nature of wireless channels, we can easily intercept.

References

[1]. F. Akyildiz and I. H. Kasimoglu, "wireless sensor and actuator network: research challenges: Adhoc networks, vol 2, No. 4, pp. 351-367, oct 2004.

[2]. "Environmental and temperature monitoring", centrak.

[3]. J. W. Hui and D. Culler, "The dynamic behavior of a data dissemination protocol for network programming at scale," in Proc. 2nd Int. Conf. Embbeded Netw. Sensor Syst., 2004, pp.81-94.

[4]. D. He, C. Chen, S. Chan, and J. Bu, "DiCode: DoS-resistant and distributed code dissemination in wireless sensor networks," IEEE Trans. Wireless Commmun., vol. 11, no. 5, pp. 1946-1956, may 2012.

[5]. 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.

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

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

[8]. M. ceriotti, G. P. Picco, A. L. Monitoring heritage buildings with wireless sensor networks: The Torre Aquila development," in Proc.IEEE Int. Conf. Inf. Process. Sensor Netw., 2009, pp.277-288.

[9]. D.He,S.Chan,S Tang, and M. Guizani, "Secure and distributed 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.

[10]. M Rahman, N. Nasser, and T. Taleb, ''Pairing-based secure timing synchronization for heterogeneous sensor networks," in Proc. IEEE Global Telecommun. Conf., 2008, pp. 1-5.

[11]. NOPP.[Online]. Available: http://www.epa.gov/geoss/

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

[13]ORION.[Online]. Available: http://www.joiscience.org/ocean observing/ advisor

[14]. P.Levis, N. Patel, D. Culler, and S. Shenker, "Trickle: A self regulating algorithm for code maintenance and propogation in wireless sensor networks," in Proc. 1st Conf. Symp. Netw. Syst. Design Implementation, 2004, pp. 15-28

[15]. A. perrig, R. Canetti, D. Song, and J. Tygar, "efficient and secure source authentication for multicast," in Proc. Netw. Distrib. Syst. Security symp., 2001, pp. 35-46.

[16]. Y Chen, I. Lin, C. Lei, and Y. Liao, "Broadcast authentication in sensor networks using compressed bloom filters," in Proc. 4th IEEE Int. Conf. Distrib. Comput. Sensor Syst., 2008, pp. 99-111.

[17]. R. Merkle, "Protocols for public key cryposystems," in Proc. IEEE security privacy, 1980, pp. 122-134.

[18]. M. Bellare and P. Rogaway,"Collision-resistant hashing: Towards making UOWHFs practical," in Proc. Adv. Cryptology, 1997, pp. 56-73.

[19]. A. Perrig, R. Canetti, J. Tygar, and D. Song, "Efficient authentication and signing of multicast streams over lossy channels," in Proc. IEEE Security Privacy, 2000, pp. 56-73.

[20]. Open SLL.[Online]. Available: http://www.openssl.org

[21]. A. Liu and P. Ning, "TinyECC: A configurable library for elliptic curve cryptography in wireless sensor networks," in Proc. ACM/IEEE Inf. Process. Sensor Netw., 2008, pp. 245-256.