DiDrip: A Secure and Distributed Protocol for Updation and Dissemination of Data in WSN

Shobhna Vedprakash Pandey [1], Dr. K. Srujan Raju [2]

[1] M.Tech Student,CMR Technical Campus, Kandlakoya(V), Medchal(M),Ranga Reddy District, Telangana, India
[2] HOD, CSE Department,CMR Technical Campus, Kandlakoya(V), Medchal(M),Ranga Reddy District, Telangana, India

Abstract— *A data discovery and dissemination protocol for wireless sensor networks (WSNs) is responsible for updation of configuration parameters and distribution of management commands to the sensor nodes. The existing data discovery and dissemination protocols faces several drawbacks. The idea behind the project is to use the first secure and distributed data discovery and dissemination protocol named DiDrip for WSN. DiDrip allows the network owners to authorize multiple network users with different privileges to directly and simultaneously disseminate data items to the nodes. Extensive security analysis shows that DiDrip is probably secure.*

Keywords— Data Dissemination, DiDrip, Wireless Sensor Networks, Zero Protocol.

I. INTRODUCTION

Wireless Sensor Network (WSN) which is deployed, usually need to update buggy/old small programs or parameters stored in the sensor nodes. This can be achieved by the so-called data discovery and dissemination protocol, which facilitates a source to inject small programs, commands, queries, and configuration parameters to sensor nodes [1]. The sensor nodes could be distributed in any harsh environment, remotely disseminating such small data to the sensor nodes through the wireless channel is a more preferred and practical approach than manual intervention. Considering the upper sub-figure in fig.1, all existing data discovery and dissemination protocols follows centralized approach, where data items can only be disseminated by the base station [2]. But the centralized approach is inefficient, non-scalable, and vulnerable to security attacks that can be launched anywhere along the communication path. Different number of security attacks may occur in Wireless sensor network that is clone attack, man in the middle attack and replay attack. Zero knowledge protocol is an improvement on these attacks.

1.1 Zero knowledge Protocol: The objective of this protocol is to have such system in which prover has to convince a verifier that he has knowledge of a certain secret without disclosing any information. In Zero Knowledge Protocols, the first party domain has the knowledge of some "secret" or private key information where he has to be verified by a second party without imparting the actual secret information or private key to that second party or to any eavesdropping third party. The first party owning the secret information or private key ("s") and trying to prove that it has possession of the information will be referred to as the "prover" ("P"); the second party wishing to verify without actually receiving knowledge of the secret will be referred to as the "verifier" ("V"). The secret information may be any numeric value, hereafter referred to as the secret number of the prover [3]. Zero knowledge protocols require less computational power, less bandwidth, and less memory compared to other authentication methods. With attachment of unique fingerprint to each node the clone attack can be addressed. Zero knowledge Protocol is helpful for preventing man-in-the middle attack and replay attack.

1.2 Algorithm for Zero knowledge Protocol [4]

1. Using Specific code find the fingerprint (S) which is used as a private key for each node.
2. Base station will maintain product of two large primes which value of N as public key.
3. The base station generates a secret code by using i.e. v= s2modN. Prover ("P") node will try to verify with Verifier ("V"). The value of v (secret code) is given to the Verifier on its request.
4. This secret code will be changing for every authentication process. The change of bits will be done diagonally. This will save memory and computation power which is important in sensor nodes.
5. Now zero knowledge protocol is applied. Some challenges are asked by Verifier ("V") to Prover ("P") based upon its secret.
6. Prover ("P") will answer the challenges but will not open secret.
7. In this case, the value of secret s is not revealed anywhere during communication and thus it will not be used by any attacker in node.

For such networks, data dissemination is better to be carried out by using technique of distribution by authorized network users. Especially in such emergent context of shared sensor networks, distributed data discovery and dissemination is an increasingly relevant matter in WSNs [5]. For example, large scale sensor networks are built in recent projects such as Geoss, NOPP and ORION. It is expected that network owners and many different users may have different privileges of dissemination. In this context, distributed operation by networks owners and users with different privileges will be a crucial issue, for which efficient solutions are still missing.

II. OBJECTIVE OF RESEARCH

There was a need of distributed data discovery and dissemination protocols for dissemination of data items to WSN, where previously proposed protocols did not address this need. After studying, the functional requirements of such protocols and after identifying their design objectives and the security vulnerabilities in previously proposed protocols. Based on the above requirements, we are using DiDrip (distributed data discovery and dissemination protocol) which allows network owners and authorized users to disseminate data items into WSNs without relying on the centralized approach [2]. Fig.1 shows the variations in centralized and distributed data discovery and dissemination approaches [1].

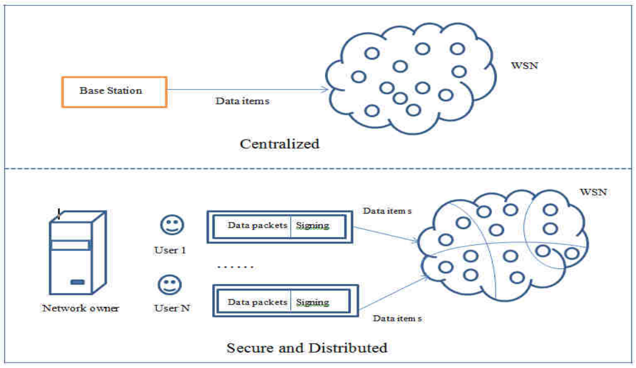


Fig.1: System overview of centralized and distributed data discovery and dissemination approaches.

III. LITERATURE REVIEW

3.1 The Dynamic Behavior of a Data Dissemination Protocol for Network Programming at Scale: To support network programming, it presents Deluge, a reliable data dissemination protocol for propagating large data objects from one or more source nodes to many other nodes over a multi-hop, wireless sensor network. Deluge builds from prior work in density-aware, epidemic maintenance protocols [6]. Using both a real-world deployment and simulation, that shows that Deluge can reliably disseminate data to all nodes and characterize its overall performance. On Mica 2- dot nodes, Deluge can push nearly 90 bytes/second, one ninth the maximum transmission rate of the radio supported under TinyOS. Control messages are limited to 18% of all transmissions. At scale, the protocol exposes interesting propagation dynamics only hinted at by previous dissemination work. A simple model is also derived which describes the limits of data propagation in wireless networks. Finally, we argue that the rates obtained for dissemination are inherently lower than that for single path propagation. It appears very hard to significantly improve upon the rate obtained by Deluge and we identify establishing a tight lower bound as an open problem.

3.2 Seluge: Secure and DoS-Resistant Code Dissemination in Wireless Sensor Networks: Wireless sensor networks are considered ideal contenders for a wide range of applications, such as industry monitoring, data acquisition in hazardous environments, and military operations. It is required and sometimes obligatory to reprogram sensor nodes through wireless links after deployment, due to, for example, the need of removing bugs and adding new functionalities [7]. The process of propagating a new code image to the nodes in a wireless sensor network is referred to as code dissemination. Seluge is a secure extension to Deluge, an open source, state of-the-art code dissemination system for wireless sensor networks. It provides security protections for code dissemination, including the integrity protection of code images and immunity from, to the best of our knowledge, all DoS attacks that exploit code dissemination protocols. Seluge is superior to all previous attempts for secure code dissemination, and is the only solution that seamlessly integrates the security mechanisms and the Deluge efficient propagation strategies.

3.3 Design of an application-cooperative management system for wireless sensor networks: SNMS is designed to be simple and have minimal impact on memory and network traffic, while remaining open and flexible [7]. The system is assessed in light of issues derived from real deployment experiences. SNMS provides a core set of services to enable management: query-based health data collection and persistent event logging. These services occupy a minimal amount of RAM and code size, and can be rapidly integrated into TinyOS applications. To ensure that these services are usable for management and will continue to function in the event of application failure, SNMS also includes new lightweight network architecture for collection and dissemination. Finally, SNMS provides a number of specific management components for selective inclusion into sensor network applications. The networking layer has been shown to perform acceptably, and the management functionality meets several needs derived from prior TinyOS sensor network deployments.

3.4 Efficient and Secure Source Authentication for Multicast: Source authentication or enablement of receivers of multicast data to verify the origin data is one of the major challenges in securing multicast communication. In cases of common settings where receivers of the data are not trusted, and where lost packets are not retransmitted, this problem becomes more complex. Several source authentication schemes for multicast have been suggested in the past, but none of these schemes is satisfactorily efficient in all prominent parameters. Recently, a new efficient scheme is proposed, TESLA which is based on initial loose time synchronization between the sender and the receivers, followed by delayed release of keys by the sender. There are possibilities of several substantial modifications and improvements to TESLA. One modification allows receivers to authenticate most packets as soon as they arrive. Other modifications improve the scalability of the scheme, reduce the space overhead for multiple instances, increase its resistance to denial-of-service attacks, and more. The basic TESLA protocol has the following salient properties are low communication overhead and perfect loss robustnessshould be flush left, and subsequent paragraphs should have a five-space indentation. A colon is inserted before an equation is presented, but there is no punctuation following the equation. All equations are numbered and referred to in the text solely by a number enclosed in a round bracket (i.e., (3) reads as "equation 3"). Ensure that any miscellaneous numbering system you use in your paper cannot be confused with a reference [4] or an equation (3) designation.

IV. ANALYSIS

Many existing Data Discovery and Dissemination protocols used for easy updation of old small programs or parameters stored in sensor nodes after the wireless sensor network is deployed. The existing data discovery and dissemination protocols namely DIP, Drip has been proposed for WSNs. The proposed protocols assume that the WSN’s operating environment is trustworthy. But in reality this is impossible because adversaries exist and the threats are imposed to affect the normal operation of WSNs [8]. The existing data discovery and dissemination protocols are more over based on centralized approach. All above existing protocols uses Trickle as a base algorithm. Trickle requires each node to periodically broadcast a summary of its stored data.

Limitations of Existing Protocol:

- In existing protocol, adversaries can launch attacks to harm the network that means security is less.
- When the connection between base station and node is broken, the data discovery and dissemination is impossible.
- Distributed operation by networks owners and users with different privileges will be a crucial issue, for which efficient solutions are missing.

The proposed system has the first secure and distributed data discovery and dissemination protocol named DiDrip. It allows the network owners to authorize multiple network users with different privileges to simultaneously and directly disseminate data items to the sensor nodes without relying on base station. DiDrip satisfies the security requirements of the protocols.

Advantages of Proposed System:

- Improved security mechanisms and it is scalable
- Multiple authorized users allowed disseminating data items into wireless sensor networks without depending on base station.
- WSNs are protected by Denial of Service (DoS) attacks.

4.1 Software Requirements:

Operating System	: Windows XP
Programming Language	: Advanced Java
Backend Database	: MySQL Command Line Client

V. IMPLEMENTATION

The primary challenge of WSNs in providing security is the limited abilities of sensor nodes in terms of computation, bandwidth, power supply, energy and storage. Commonly used solution is digital signature to provide authentication function for dissemination of data. That is, users digitally sign each packet individually and nodes need to verify the signature before processing it. Another possible approach is by symmetric key cryptography. But, this approach is because once a node is compromised, the globally shared keys will be revealed. So we choose digital signatures over other forms for update packet authentication. DiDrip is based on elliptic curve cryptography (ECC). Considering the lower sub-figure in fig.1, we will develop DiDrip which has following four phases [2]:

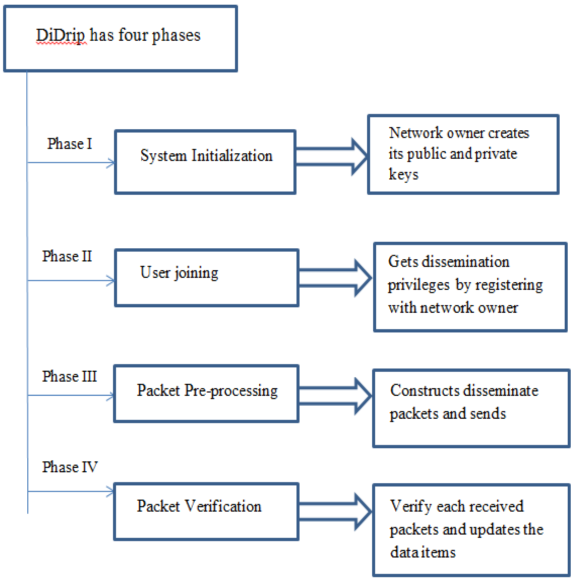

DiDrip has four phases
Phase I
System Initialization
Network owner creates its public and private keys
Phase II
User joining
Gets dissemination privileges by registering with network owner
Phase III
Packet Pre-processing
Constructs disseminate packets and sends
Phase IV
Packet Verification
Verify each received packets and updates the data items

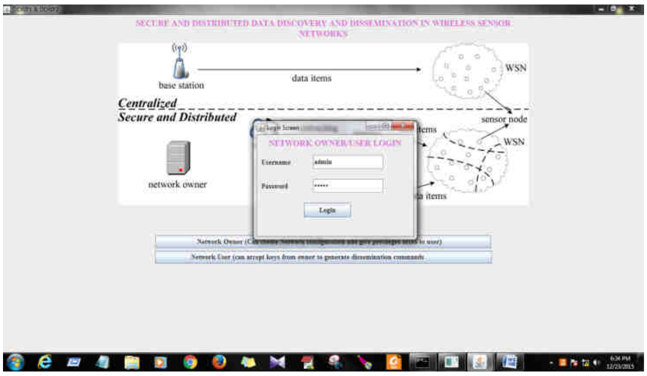
Fig. 2: Phases of DiDrip

1. System Initialization: The network owner has to carry out following steps for computing its public key that can be done by choosing its own private key and defining public parameters which can be used to generate public keys for communications. Before the network deployment, it loads the public parameters in each node for user purpose. And it uses its private key for verification of the packets.
2. User joining: Before starting dissemination of data, user has to register with network owner by giving its identity and other details. In user joining phase, network owner will allow him/her to send and/or receive data to/from only particular nodes by providing him/her privilege access. User chooses private key and computes public keys using public parameters which was loaded by network owner. Many users can register at the same time for communication purposes and there details are stored in the backend database.
3. Packet preprocessing: In packet preprocessing phase, construction of packets needs to be done for dissemination of message or commands. The packets are constructed and encrypted by using encryption technique for communication. In our research, we will be using commands for dissemination of data. After issuing commands, simulation graph gets started to show how user accesses its privileged nodes in the network to disseminate the data. Readings are captured from each node which disseminates the data to each other and it is displayed.
4. Packet verification: In the packet verification phase, verification of each received packet is done by every node. It updates the data in the node from each received packet, if the verification result is positive. Finally after verification, we developed resource graphs which can be used to see the execution time of the command and the memory space used by the nodes for dissemination. We are going to use Mysql line command software used as database for storing the registered users credentials and privileges. The different security requirement is satisfied by DiDrip in research, they are: Distributed; Supporting different user privileges; Authenticity and integrity of data items; User accountability.

But still, DiDrip authentication is vulnerable to DoS attacks. To prevent the network owner from impersonating users, user certificates are issued by a certificate authority of a public key infrastructure (PKI). But every time generation, transmission and verification of certificates of each user is not efficient. We can use ZKP which allows one party to prove its knowledge of a secret to another party without ever revealing the secret. Authentication systems motivates all the research of zero knowledge proofs in which prover wants to prove its identity to a verifier through some secret information (such as a password) but never wants that the second party to get anything about this secret. This known as "zero-knowledge proof” [9].

VI. RESULTS AND DISCUSSION

We are going to show the results by using simulation graph, where owner has to register many different users with different privileges. Registering user with its name and password and providing the privileges to access for particular nodes for example, user has privilege to access only 0-5 numbered nodes. Public keys are computed for user and a network with multiple nodes is generated. To login as an owner, click on “Network Owner” and enter user name and password as “admin”.



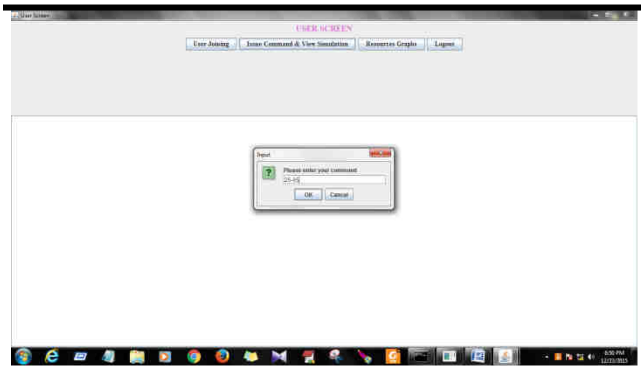
Username admin

Password •••••

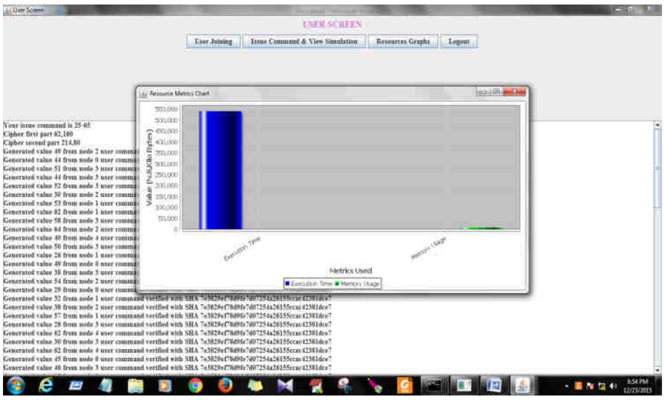
Network Owner (Can create Network configuration and give privileges to user)

Network User (can accept keys from owner to generate dissemination commands

Owner can register users with privileges by clicking on “Register User With Privileges”.

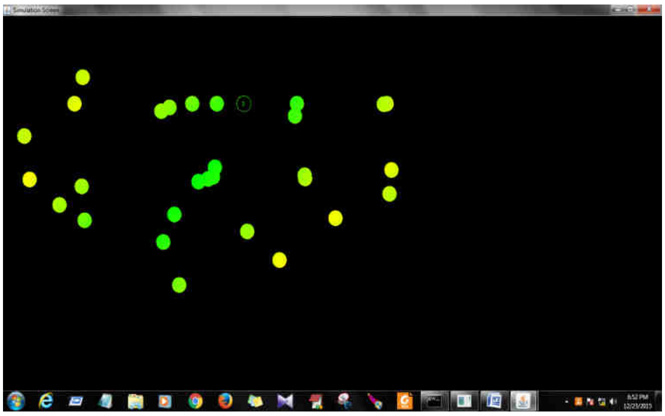


To issue commands to the nodes click on "Issue command and view simulation" and enter readings like 25-65. And the captured values are displayed on the user screen. Only privileged nodes can disseminate the data and received data is verified by node and then updation takes place in the node.



```
Your issue command is 25-65
Cipher first part 02,100
Cipher second part 214,88
Generated value 49 from node 2 user comma
Generated value 44 from node 0 user comma
Generated value 51 from node 3 user comma
Generated value 44 from node 3 user comma
Generated value 52 from node 3 user comma
Generated value 30 from node 2 user comma
Generated value 53 from node 1 user comma
Generated value 02 from node 1 user comma
Generated value 58 from node 3 user comma
Generated value 64 from node 2 user comma
Generated value 40 from node 4 user comma
Generated value 50 from node 3 user comma
Generated value 26 from node 1 user comma
Generated value 49 from node 0 user comma
Generated value 38 from node 3 user comma
Generated value 54 from node 2 user comma
Generated value 29 from node 0 user comma
Generated value 32 from node 1 user command verified with SHA 7e3829ef78d9fe7d07254a26155ccac42381dce7
Generated value 38 from node 2 user command verified with SHA 7e3829ef78d9fe7d07254a26155ccac42381dce7
Generated value 57 from node 1 user command verified with SHA 7e3829ef78d9fe7d07254a26155ccac42381dce7
Generated value 28 from node 3 user command verified with SHA 7e3829ef78d9fe7d07254a26155ccac42381dce7
Generated value 62 from node 4 user command verified with SHA 7e3829ef78d9fe7d07254a26155ccac42381dce7
Generated value 30 from node 3 user command verified with SHA 7e3829ef78d9fe7d07254a26155ccac42381dce7
Generated value 62 from node 4 user command verified with SHA 7e3829ef78d9fe7d07254a26155ccac42381dce7
Generated value 45 from node 0 user command verified with SHA 7e3829ef78d9fe7d07254a26155ccac42381dce7
Generated value 46 from node 3 user command verified with SHA 7e3829ef78d9fe7d07254a26155ccac42381dce7
```

Resource graphs used to see the execution of the command and the memory space used.



See the simulation graph, where the empty node represents generating readings in nodes when command is entered by user. User credentials and privileges can be seen in Mysql line command software which acts as backend database.

VII. CONCLUSION AND FUTURE WORK

Different security vulnerabilities are identified in existing data discovery and dissemination when used in WSNs [10]. Also, none of these approaches support distributed operation. Therefore, a secure and distributed data discovery and dissemination protocol named DiDrip has been proposed. The authenticity and integrity of the disseminated data items in DiDrip is preserved. Also, due to the open nature of wireless channels, messages can be easily intercepted.

Thus, in the future work, we will consider how to ensure data confidentiality to make more secure and distributed data discovery and dissemination protocols. To enhance the protocol, two modifications are to be done to improve the security and efficiency of DiDrip.

1. Avoiding the Generation, Transmission and Verification of Certificates.
2. Message Specific Puzzle Approach for Resistance to DoS attacks.

REFERENCES

[1] Daojing He, Sammy Chan, Shaohua Tang and Mohsen Guizani, "Secure Data Discovery and Dissemination based on Hash Tree for Wireless Sensor Networks", IEEE transactions on wireless communications, Vol. 12, No. 9, September 2013.

[2] Pandey, Shobhna Vedprakash, and K. Srujan Raju. "Secure and Efficient DiDrip Protocol for Improving Performance of WSNs." International Journal of Advanced Engineering, Management and Science (IJAEMS) (2016).

[3] Parbat, Vishal, et al. "Zero knowledge protocol to design security model for threats in WSN." *Int. J. Eng. Res. Appl.(IJERA)* 2 (2012): 1533-1537.

[4] Sagar Dhawale, B G Hogade. "Secured Wireless Sensor Network by Using Zero Knowledge Protocol." International Journal of Advance Foundation And Research In Science & Engineering (2015).

[5] Mohammad A. Matin, Wireless Sensor Networks: Technology and Protocols: Published by InTech, Croatia, ISBN 978-953-51-0735-4, 2012.

[6] Jisha Mary Jose, Jomina John, "Data dissemination protocols in wireless sensor networks-a survey", IJARCCE, March 2014.

[7] G. Tolle and D. Culler, "Design of an application cooperative management system for wireless sensor networks," in Proc. EWSN, pp. 121–132, 2005.

[8] Krontiris, Ioannis, et al. "Cooperative intrusion detection in wireless sensor networks." *European Conference on Wireless Sensor Networks*. Springer Berlin Heidelberg, 2009.

[9] D'yachkov, Arkadii G., and Vyacheslav V. Rykov. "Optimal superimposed codes and designs for Renyi's search model." *Journal of Statistical Planning and Inference* 100.2 (2002): 281-302.

[10] Salvatore La Malfa, Wireless Sensor Networks, 2010.