International Journal of Engineering Research and General Science Volume 4, Issue 2, March- April, 2016
ISSN 2091-2730

SURVEY ON SECURE AND DISTRIBUTED DATA DISCOVERY AND DISSEMINATION IN WSN

B.Gouthami

M. Tech Student (GNITS, Hyderabad),gouthamichinni.01743@gmail.com, 9440022281.

A.Naveena

Assistant Professor(GNITS, Hyderabad), ambidinaveena@yahoo.com,9866937347

Abstract — Dissemination is a basic sensor network protocol. The ability to reliably deliver a piece of data to every node allows administrators to reconfigure, query, and reprogram a network. Data Discovery and Dissemination Protocol is responsible for updating the configuration small commands, parameters, queries, variables. Dissemination protocols have self-organizing capabilities because WSNs are deployed in critical environments and remote areas where manual reprogramming of nodes are difficult. All existing data discovery and dissemination protocols suffer from two drawbacks. First, they are based on centralized approach and second, they were not designed with security in mind and hence adversaries can easily launch attacks. This paper proposes DiDrip protocol. This is the first secure and distribute Data discovery and dissemination protocol.

Keywords—Data Discovery, Data dissemination**,** wireless sensor networks, reprogramming, Trickle, tuples, security

INTRODUCTION

A wireless sensor network (WSN) is made of a group of nodes and is used for monitoring and analysis purposes. Wireless sensor networks mainly use broadcast communication.In comparison with wired systems, wireless sensor networks have their own special features and limitations. Wireless sensor networks are limited by sensors limited power, energy and computational capability. The evolving conditions and environment can change application requirements. This requires altering the behavior of the network by introducing new code or updates. This can be achieved by data discovery and dissemination protocol, which facilitates a source to inject small programs, commands, queries, and configuration parameters to sensor Nodes [1]. Unlike flooding protocols, which are discrete efforts that terminate, possibly not delivering the data to some nodes, dissemination achieves reliability by using a continuous approach that can detect when a node is missing the data. Reliability is important because it makes the operation robust to temporary disconnections or high packet loss.

Some WSNs do not have any base station at all. For example, for a WSN monitoring human trafficking in a country's border or a WSN deployed in a remote area to monitor crop cultivation, a base station becomes an attractive target to be attacked. For such networks, data dissemination is better to be carried out by authorized network users in a distributed manner.

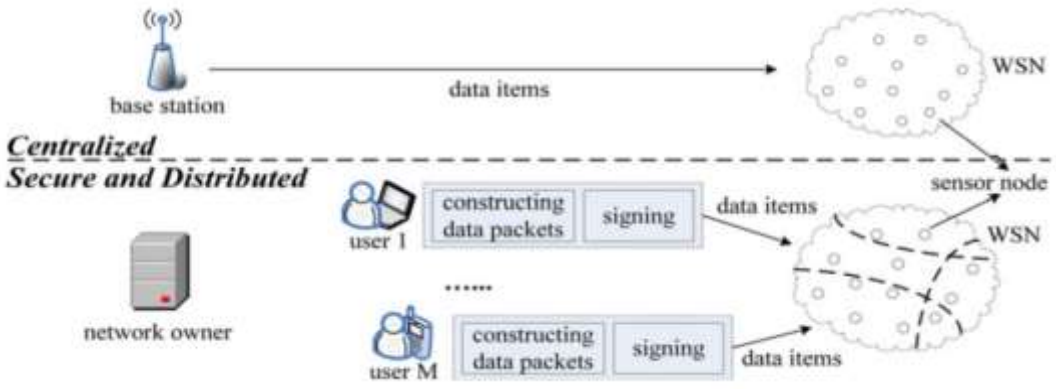

Figure 1: shows the difference between centralized and distributed dissemination of data packets.

Fig.1, data items can only be disseminated by the base station. Unfortunately, this approach suffers from the single point of failure as dissemination is impossible when the base station is not functioning or when the connection between the base station and a node is broken. In addition, the centralized approach is inefficient, non-scalable, and vulnerable to security attacks that can be launched anywhere along the communication path .Even worse, some WSNs do not have any base station at all. For example, for a WSN monitoring human trafficking in a country's border or a WSN deployed in a remote area to monitor illicit crop cultivation, a base station becomes an attractive target to be attacked. For such networks, data dissemination is better to be carried out by authorized network users in a distributed manner.

The primary challenge of providing security functions in WSNs is the limited capabilities of sensor nodes in terms of computation, energy and storage.

LITERATURE SURVEY:

1. TRICKLE: A Self Regulating Algorithm

Propagating code is costly; learning when to propagate code is even more so. Motes must [1] periodically communicate to learn when there is new code. To reduce energy costs, motes can transmit metadata to determine when code is needed. Trickle, an algorithm for propagating and maintaining code updates in wireless sensor networks. It is based on "Polite Gossip".

Each node only gossip about new things that it has heard from its neighbors, but it won't repeat gossip it has already heard, as that would be rude. Each mote periodically broadcasts metadata describing what code it has. However, if a mote hears gossip about identical metadata to its own, it stays quiet. When a mote hears old gossip, it triggers a code update, so the gossiper can be brought up to date. To achieve both rapid propagation and a low maintenance overhead, motes adjust the length of their gossiping attention spans, communicating more often when there is new code.

For example, if mote A broadcasts that it has code $\emptyset_x$,but B has code $\emptyset_{x+1}$,then B knows that A needs an update. Similarly, if B broadcasts that it has $\emptyset_{x+1}$, A knows that it needs an update. If B broadcasts updates, then all of its neighbors can receive them without having to advertise their need. Some of these recipients might not even have heard A's transmission. In this example, it does not matter who first transmits, A or B; either case will detect the inconsistency. [2]All that matters is that some motes communicate with one another at some nonzero rate; we will informally call this the "communication rate." As long as the network is connected and there is some minimum communication rate for each mote, everyone will stay up to date. The fact that communication can be either transmission or reception enables Trickle to operate in sparse as well as dense networks.

Each mote maintains a counter c, a threshold k, and a timer t in the range [0, T]. K is a small, fixed integer and T is a time constant. When a mote hears metadata identical to its own, it increments c. At time t, the mote broadcasts its metadata if c < k. When the interval of size T completes, c is reset to zero and t is reset to a new random value in the range [0, T].Scaling logarithmically with density, it can be used effectively in a wide range of networks [2]. One limitation of Trickle is that it currently assumes motes are always on. To conserve energy, long-term mote deployments often have very low duty cycles (e.g., 1%).Correspondingly, motes are rarely awake, and rarely able to receive messages.

Advantages:

i. Can be used effectively in wide range of networks.
ii. Trickle can scale to very dense networks.

Limitations:

i. Sensor nodes are always on
ii. Overhead increases with number of data items

2 DRIP

SNMS [4], a Sensor Network Management System. SNMS is designed to be simple and have minimal impact on memory and network traffic, while remaining open and flexible. The system is evaluated in light of issues derived from real deployment experiences**.** In

contrast, wireless sensor networks act in aggregate, and thus, a wireless sensor network management system must be able to manage in the aggregate. Aggregate management requires a dissemination protocol that can deliver messages reliably to a set of nodes within a sensor network. The underlying algorithm used by our dissemination layer is the Trickle algorithm.

Trickle [2] uses periodic retransmissions to ensure eventual delivery of the message to every node in the network. To minimize the number of required messages, retransmissions can be suppressed by prior transmissions of similar messages, and randomization is used to prevent permanent suppression. [4] The dissemination layer takes the Trickle retransmission algorithm and builds a transport-layer interface atop it. The SNMS dissemination protocol, named Drip, provides a transport layer interface to multiple channels of reliable message dissemination. Drip provides a standard message reception interface. Each component wishing to use Drip registers a specific identifier, which represents a reliable dissemination channel. Messages received on that channel will be delivered directly to the component. Each node is responsible for caching the data extracted from the most recent message received on each channel to which it subscribes, and returning it in response to periodic rebroadcast requests. Drip avoids redundant transmission and achieves greater efficiency.

Advantages:

i. Avoids redundant transmission and greater efficiency.
ii. Gives information about health of nodes.

Limitations:

i. Centralized method.
ii. Security is not provided for data.

3 DIP

DIP, a data discovery and dissemination protocol for wireless networks [5]. Prior approaches, such as trickle or [3] SPIN, have overheads that scale linearly with the number of data items. For T items, DIP can identify new items with O (log (T)) packets while maintaining an O (1) detection latency.

DIP is a hybrid data detection and dissemination protocol. It separates this into two parts: detecting that a difference occurs, and identifying which data item is different.DIP dynamically uses a combination of searching and scanning based on network and version metadata conditions. [5] To aid its decisions, DIP continually estimates the probability that a data item is different. DIP maintains these estimates through message exchanges. When probabilities reach 100%, DIP exchanges the actual data items. It is an eventual consistency protocol in that when data items are not changing, all nodes will eventually see a consistent set of data items.

DIP stores a version number for each data item. DIP periodically broadcast a summary message containing hashes of its keys and versions. Hash-tree based algorithm detects if there is an update. If a node hears a hash that differs from its own, it knows that a difference exists. On receiving a data message whose version number is newer than its own, DIP installs the new item. DIP improves searching performance by combining hashes over ranges of the key space with a bloom filter. The overhead problem is reduced and this protocol sends 20-60% fewer packets than previous protocols.

Advantages:

i. Overhead decreases.
ii. Sends 20-60% fewer packets than existing protocols**.**

Limitations:

i. No security system.
ii. Centralized system.

3 DHV

It is a code consistency maintenance protocol (Difference detection, Horizontal search, and Vertical search). The main objective is to overcome the disadvantages of previous protocols like DRIP [4] and DIP [5] by reducing the complexity. Here data items are represented as tuples (key, version).If two versions are different; they may only differ in a few least significant bits of their version number rather than in all their bits.

DHV [6] includes two important phases: Detection phase and Identification phase. In detection, each node will broadcast a hash of all its versions in SUMMARY message .In identification, the horizontal search and vertical search steps are used. In horizontal search, a node broadcasts a checksum of all versions, called a HSUM message. In vertical search, the node broadcasts a bit slice, starting at the least significant bit of all versions, called a VBIT message After identifying this, the node broadcasts those (key, version) tuples in a VECTOR message.

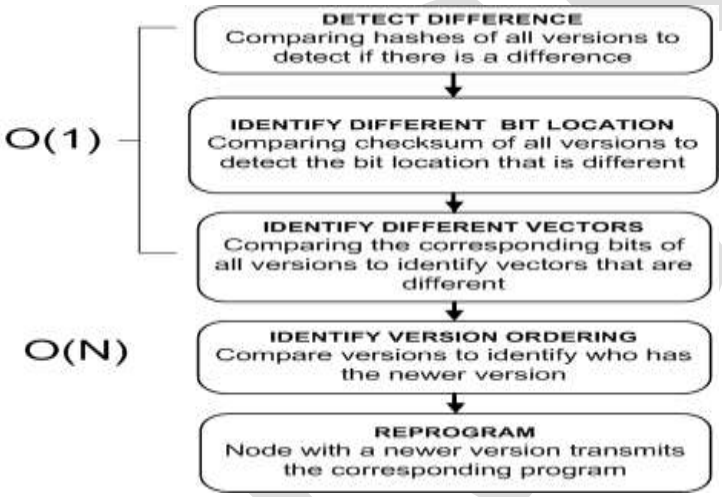

Figure 2: Main steps in the DHV protocol

Advantages:

i. DHV reduces the number of transmitted bytes.
ii. DHV performs at least twice better than DIP protocol.

Limitations:

i. Centralized approach.
ii. Chance for DOS attack.

5 Code drip

Code Drip [7] utilizes Network Coding to improve energy efficiency, reliability, and speed of dissemination. Network coding allows recovery of lost packets by combining the received packets thereby making dissemination robust to packet losses. Code Drip uses Network Coding to improve the efficiency of dissemination in Wireless Sensor Networks. Network Coding is a technique that combines packets in the network thereby increasing the throughput, decreasing energy consumption, and reducing the number of messages that are transmitted. Dropped packets are recovered using re-transmissions. By combining packets using network coding, it is possible to re- cover the transmitted information without needing to retransmit all the lost packets to all the nodes.

Network Coding allows packets to be combined using a XOR logic operator. Like Drip, Code Drip uses the Trickle timer [2] to time the message transmissions with the goal that the data will eventually arrive at all the nodes in the network. The unmodified Drip message has

only one identifier and its payload is the data to be disseminated. A combined message has two or more identifiers corresponding to the packets that were combined.

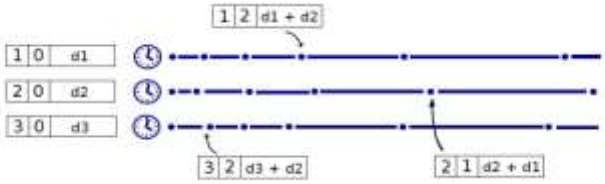

Figure 3: Code Drip example. There are three values to be disseminated. Each value has an associated Trickle timer. Each packet transmission might combine the packets.

Code Drip is faster than Drip, DIP and DHV protocols to disseminate information. It also requires less ROM memory than Drip, DHV and DIP. Code Drip is faster, smaller and sends fewer messages than Drip, DHV and DIP protocols.

Advantages:

i. Applying Network Coding reduces the dissemination time.
ii. Packet loss is reduced.

Limitations:

i. Centralized method.
ii. Need maximum buffer size.

6 DiDrip

DiDrip consists of four phases, system initialization, user joining, and packet pre-processing and packet verification. For our basic protocol, in system initialization phase, the network owner creates its public and private keys, and then loads the public parameters on each node before the network deployment. In user joining phase, a user gets the dissemination privilege through registering to the network owner. In packet pre-processing phase, if a user enters to the network and wants to dissemination some data items, he/she will need to construct the data dissemination packets and then send them to the nodes. In packet verification phase, a node verifies each received packet. If the result is positive, it updates the data according to the received packet. Based on the design objectives, they propose DiDrip. It 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.

CONCLUSION

In this paper, a survey has been done on various existing dissemination protocols for wireless sensor networks. We have seen the architecture and security considerations on each protocol. In wireless sensor networks, security is the key objective in dissemination of data. Therefore in this paper, a secure and distributed data discovery and dissemination protocol named DiDrip has been proposed. Di-Drip protocol addresses many of the drawbacks and limitations of the previous dissemination protocols like security, distributed approach, multi owner-multi user capabilities but its efficiency and security is tested under limited parameters.

REFERENCES:

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

[2] P. Levis, N. Patel, D. Culler, and S. Shankar. "Trickle: A self-regulating algorithm for code maintenance and propagation in wireless sensor networks." In First USENIX/ACM Symposium on Network Systems Design and Implementation (NSDI), 2004

[3] I.F. Akyildiz, W. Su, Y. Sankarasubranmaniam, and Erdal Cayirci, "A Survey on Sensor Networks", IEEE Communications Magazine, November 7, 2002, pp. 102-114, vol. 40(8).

[4] G.Tolle, D.Culler "Design of an application-cooperative management system for wireless sensor networks". In: Proceedings of the 2005 International Conference on Information Processing in Sensor Networks (IPSN 2008),, IEEE Computer Society (2005) 121 - 132.

[5] Lin, K., Levis, P.: "Data discovery and dissemination with dip". In: Proceedings of the 2008 International Conference on Information Processing in Sensor Networks (IPSN 2008), Washington, DC, USA, IEEE Computer Society (2008) 433–444.

[6] 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.

[7] NildoRibeiro Junior, Marcos A. M. Vieira, Luiz F. M. Vieira, and Omprakash Gnawali,"CodeDrip: Data Dissemination Protocol with Network Coding for Wireless Sensor Networks", in Proceedings of the 11th European conference on Wireless sensor networks (EWSN 2014), Feb. 2014