ISSN:0975-9646

Design of Lightweight Secure and Distributed Reprogramming Protocol for Wireless Sensor Networks using Identity-Based Signature Scheme

Pooja Motwani[*], Purnima Soni[#], Priyanka Fulare[#]

[#]*Computer Science & Engineering Department, Rashtrasant Tukdoji Maharaj Nagpur University, GHRIETW, Nagpur, India*

[*]*Computer Science & Engineering Department, Rashtrasant Tukdoji Maharaj Nagpur University, GHRIETW, Nagpur, India*

***Abstract*— A novel secure and distributed reprogramming protocol called SDRP is the prime distributed reprogramming protocol for Wireless Sensor Networks (WSNs). SDRP relies on distributed reprogramming approach that permits multiple authorized network users to directly and simultaneously update program images on different sensor nodes without including the base station. SDRP expands Deluge to be a secure protocol but Deluge becomes inefficient with respect to delay, communication and energy to high network density. A new improved SDRP using identity-based short signature scheme has been designed. Rateless Deluge has many benefits as compared to Deluge such as minimizing latency at reasonable levels of packet loss, generally utilizing much less energy, and being more scalable to high network density, a major resource in WSNs. Therefore, for further improvement of the reprogramming efficiency of improved SDRP in terms of delay, communication and energy, present work is based on how to integrate improved SDRP with a better reprogramming protocol like Rateless Deluge, which leads to lightweight secure and distributed reprogramming.**

***Keywords*— Efficiency, identity-based short signature scheme, lightweight, reprogramming, wireless sensor networks.**

I. INTRODUCTION

Wireless reprogramming is the process of propagating a new program image or appropriate commands to sensor nodes in wireless sensor networks [1] - [3]. A novel secure and distributed reprogramming protocol called SDRP has been suggested, which is the prime distributed reprogramming protocol for WSNs. SDRP is based on distributed reprogramming approach that permits multiple authorized network users to directly and simultaneously update program image on different sensor nodes without including the base station (network owner). SDRP can acquire all traits of distributed reprogramming such as distributed reprogramming, user traceability, scalability, robust security, supporting different privileges for each user, and high efficiency [4]. Moreover, SDRP maintains the benefits of the well-known protocols such as Seluge [5] and Deluge [6], [7]. Furthermore, for security and efficiency point of view, any efficient identity-based signature (IBS) scheme which has been available after public scrutiny for many years can be directly used in SDRP [8]. The identity-based short signature scheme produces shortest and simplest signatures and requires less computation cost, thus, is more efficient than all well-known IBS schemes [9]. A new improved SDRP using identity-based short signature scheme has been designed recently [10]. Rateless Deluge has many benefits as compared to Deluge such as minimizing latency at reasonable levels of packet loss, generally utilizing much less energy, and being more scalable to high network density, a major resource in WSNs [11]. Therefore, for further improvement of the reprogramming efficiency of improved SDRP in terms of delay, communication and energy, integration of improved SDRP with Rateless Deluge leads to lightweight secure and distributed reprogramming.

The rest of this paper is organized as follows. Section II briefly reviews SDRP, improved SDRP, identity-based short signature scheme and rateless Deluge. Section III describes the design of improved SDRP using short IBS scheme. Section IV describes the design of lightweight SDRP. Section V shows our simulation results in terms of delay, communication and energy. The last section concludes this paper.

II. RELATED WORK

Secure and Distributed Reprogramming Protocol expands Deluge to be a secure protocol but Deluge becomes inefficient with respect to delay, communication and energy to high network density. A novel IBS scheme has been proposed for secure and distributed reprogramming in WSNs. This scheme requires two pairing operation on a sensor node [4]. The IBS algorithm by Barreto et al. [12] as an example has been chosen to show that, for efficiency and security point of view, any efficient (IBS) scheme which has been available after public scrutiny for many years can be directly used in SDRP. The size of signature used in Barreto et al. scheme is nearly equal to 320 bits [8]. The identity-based short signature scheme produces shortest and simplest signatures and requires less computation cost, thus, is more efficient than all well-known IBS scheme[13] – [15]. The size of signature used in short IBS scheme is nearly equal to 160 bits [9]. This scheme requires one pairing operation. The scheme [12] also requires one pairing computation but it wants two exponentiation operations on a cyclic multiplicative group, since the research [16] reveals the exponentiation operation on a cyclic multiplicative group is much time spending when the embedding degree is huge. Rateless Deluge uses rateless codes such as random linear codes to transmit data

so that the transfer mechanism of the original Deluge gets altered. Rateless Deluge has many benefits as compared to Deluge such as minimizing latency at reasonable levels of packet loss, generally utilizing much less energy, and being more scalable to high network density, a major resource in WSNs [17].

III. Design of Improved SDRP using Identity-Based Short Signature Scheme

A. System Initialization Phase

The network owner executes the following steps.

1) Key setup: Given a security parameter k, the network owner chooses two groups G_1 and G_2 of same prime order $q > 2^k$ and a modified Weil pairing map $e: G_1 \times G_1 \rightarrow G_2$. P is a generator of groups G_1. Let $g = e(P, P)$, then the network owner selects cryptographic hash functions $H_1 : \{0,1\}^* \rightarrow Z_q^*$, $H_2 : \{0,1\}^* \times G_1 \rightarrow Z_q^*$ and picks a random number $s \in Z_q^*$ as its master key and computes its public key $PK_{owner} = sP \in G_1$. Afterwards, the network owner publishes the system parameters $\{k, G_1, G_2, e, q, P, g, PK_{owner}, H_1, H_2\}$, but keeps s secret.

2) User public/private key generation: Consider a user Uj with identity $UIDj \in \{0, 1\}^*$ who registers to the network owner. After verifying his registration information, the network owner first sets Uj's public key as $PK_j = H_1 (UID_j \| Pri_j)$ and computes the corresponding private key $SK_j = (1/(s + PK_j)) P$ and then sends $\{PK_j, SK_j, Pri_j\}$ back to Uj by a secure channel. Here, Prij denotes the level of user privilege such that user Uj is allowed to reprogram the sensor nodes set with specified identities or/and within a particular region during his subscription period (i.e., the beginning time and the end time).

B. User Preprocessing Phase

User Uj takes the following actions.

1) Uj partitions the program image to Y fixed-size pages, represented as page 1 through page Y. Uj splits page i ($1 \le i \le Y$) into N fixed-size packets, represented as Pkti,1 through Pkti,N . A Merkle hash tree [18] is used to facilitate the authentication of the hash values of the packets in second page (page 1). The packets related to this Merkle hash tree collectively are referred as initial page (page 0). The root of the Merkle hash tree, the metadata about the program image (e.g., version number, targeted node identity set, and program image size), and a signature over all of them are included in a signature message. Assume that the message m represents the root of the Merkle hash tree and the metadata about the program image. Then, in order to ensure the authenticity and integrity of the new program image, Uj takes the following actions to build the signature message [4].

2) Before signing, Uj firstly picks a random number $r \in Z_q^*$, computes $U = rQ = r(PK_{owner} + PK_j P)$ and broadcasts U as a public parameter, and then keeps r secret. In order to generate a signature σ_j for a message $m \in \{0,1\}^*$, Uj's work as described in the following. Sets $h = H_2(m, U)$ and Computes $\sigma_j = (1/(r + h)) SK_j$. Then σ_j is the signature of Uj with identity UIDj on a message m.

3) Uj transmits to the targeted nodes the signature message {UIDj, Prij, m, σj}, which serves as the notification of the new program image. SDRP relies on the underlying Deluge protocol to distribute packets for a given program image.

C. Sensor Node Verification Phase

Upon receiving a signature message, each sensor node verifies it as follows.

1) The sensor node first pays attention to the legality of the programming privilege Prij and the message m. For example, the node needs to test whether the identity of itself is included in the node identity set of Prij. Only if they are valid, the verification procedure goes to the next step.

2) Given the system public parameters $\{k, G_1, G_2, e, q, P, g, PK_{owner}, H_1, H_2\}$ assigned by the network owner, the sensor node performs the following verification:

$e(S, U + hQ) = g$

If the equation holds, the signature σj is valid because

$$\begin{aligned} e(\sigma_j, U+hQ) &= e(1/(r + h)\, SK_j, rQ+hQ) \\ &= e(1/(r + h)\, SK_j, (r+h)Q) \\ &= e((1/(s + PK_j))\, P, (PK_{owner} + PK_j P)) \\ &= e((1/(s + PK_j))\, P, (sP + PK_j P)) \\ &= e(P, P) = g \end{aligned}$$

Otherwise, the sensor node simply drops the signature.

3) If the aforementioned verification passes, the sensor node believes that the message m and the privilege Prij are from an authorized network user with identity UIDj. Hence, the sensor node accepts the root of the Merkle hash tree constructed for initial page (page 0). Thus, the nodes can authenticate the hash packets in page 0 once they receive such packets, based on the security of the Merkle hash tree. The hash packets include the hash values of the data packets in page 1. Therefore, after verifying the hash packets, a node can easily verify the data packets in page 1 based on the one-way property of hash functions. Likewise, once the data packets in page i have been verified, a sensor node can easily authenticate the data packets in page i + 1, where i = 1, 2, . . . , Y − 1. Only if all verification procedures described previously pass, the sensor node accepts the program image.

IV. Design of Lightweight SDRP

The design of a lightweight SDRP named LSDRP involves two important parts. The first part is the design of the random linear codes as rateless codes to reduce communication, latency, and energy use. The second part involves re-engineering the improved SDRP (relies on the underlying Deluge protocol) data transfer mechanism. It means that the newly designed rateless data transfer mechanism must integrate with the improved SDRP in a natural way.

Random linear codes are rateless and permit for lightweight design so it is used for encoding and decoding process [17].

Lightweight SDRP modifies the improved SDRP in that it applies rateless codes to transmit the data. This variation

causes notable structural changes to the method for requesting and transferring data so that communication gets reduced in the control plane and the data plane. The change to the request mechanism is quite simple. The knowledge of the specific packets missed is not required by the lightweight SDRP and thus only need the number of missed packets as a single byte instead of a bit vector.

The transfer state machine must load all data, encode, and broadcast the encoded packets at the network user. Once the necessary number of encoded packets is transmitted, the next page is precoded at the source on its availability. A simplified state diagram for the new mechanism at the network user is shown in Fig. 1. This mechanism at the network user is designed in user preprocessing phase of the improved SDRP.

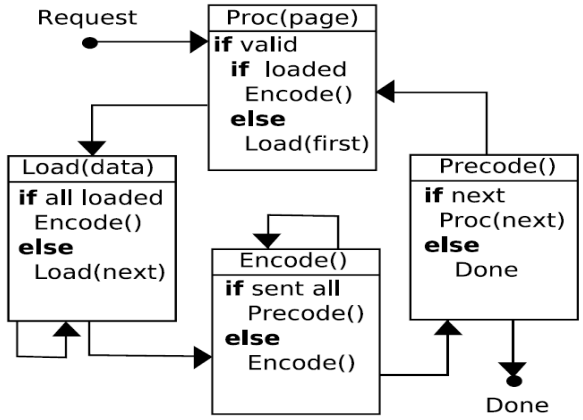


Fig. 1 State diagram at the source for a valid request for a data packet. Upon reception of the request the source loads all data, encodes, and transmits encoded packets. After transmitting the required number of encoded packets the source precodes the next page if available.

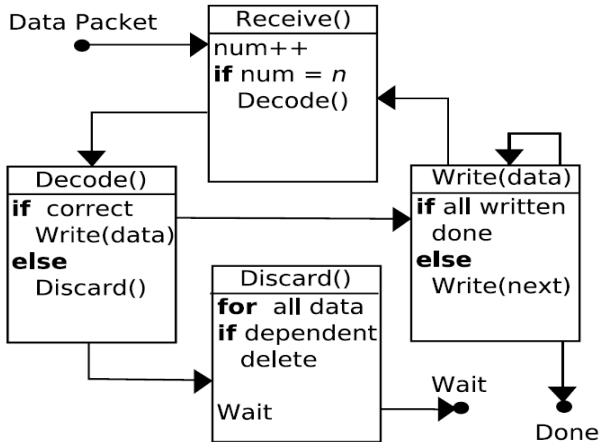


Fig. 2 State diagram at the receiving node for a valid data packet. Once the node has received *k* encoded packets it attempts to decode. If decode is successful the node writes the data to Flash. Otherwise the node discards any linearly independent packets and waits for more encoded packets.

The process for data reception at the receiving sensor nodes is changed to permit the nodes to receive k individual encoded packets and decode the page. A simplified state diagram for the new mechanism at the receiving nodes is shown in Fig. 2. This mechanism at the receiving sensor nodes is designed in sensor node verification phase of the improved SDRP.

V. Simulation Results

Lightweight SDRP, which is based on simulation, is designed in VB.Net. Table I shows simulation parameters used in the design of lightweight SDRP.

TABLE I
SIMULATION PARAMETERS

Parameter	Description / Value
Routing Protocol	Deluge, Improved SDRP
Nodes	35
Transmission Range	250m
Energy	100J
Mac Layer	802.11

As shown in Fig. 3, WSN consists of a large number of resource-constrained sensor nodes, many sensor network users, and a single network owner. Here, the network owner can be offline. Also, after the users register to the owner, they can enter the WSN and then have predefined privileges to reprogram the sensor nodes without involving the network owner. Sensor nodes perform verification and if verification passes then only accept new code image.

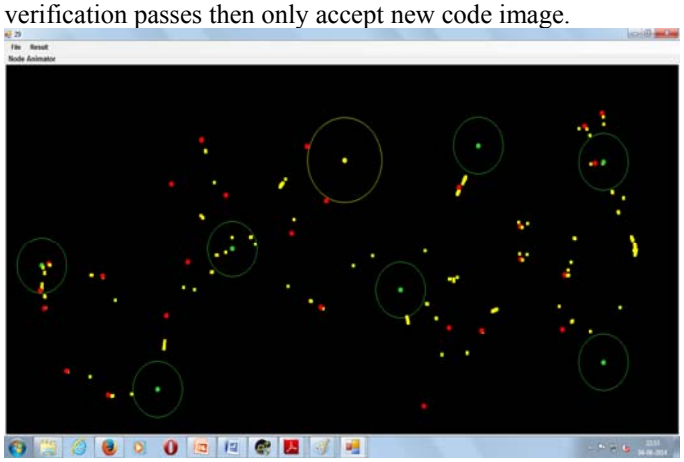


Fig. 3 System Overview of Lightweight SDRP

Trace file shown in Fig. 4, contains packet information such as source id, source ip address, destination id, destination ip address, time, energy, packet status (send, receive, reprogram, switch), packet id, packet size, and data. Trace file is used to plot graphs for delay, communication on data plane and control plane, and energy.

The following four metrics are used to compare the improved SDRP using identity-based short signature scheme with the lightweight SDRP, namely, execution time, latency, communication on the data plane and the control plane, and network lifetime.

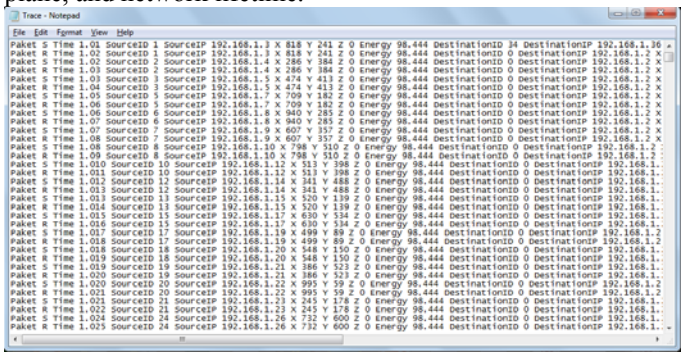
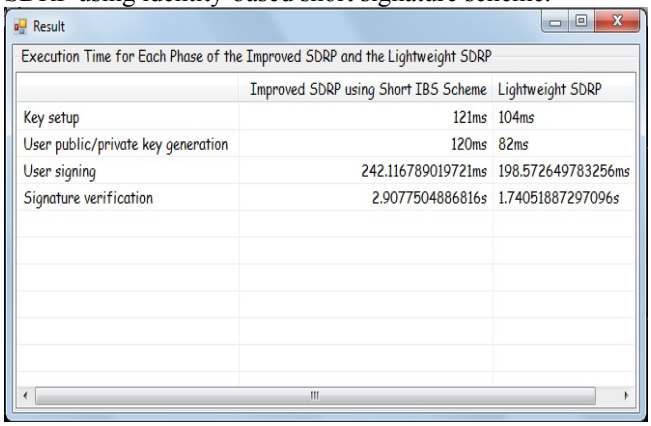


Fig. 4 Trace File

The execution time measures the time period for each operation of the two protocols. The latency measures the time to disseminate the program image. The communication on the data plane and the control plane depends upon the number of data packets and the number of request packets transmitted in the network respectively. The network lifetime mainly depends on the less energy consumption of the protocol.

Fig. 5 shows execution time for each phase of the improved SDRP using short IBS scheme and the lightweight SDRP. Our simulation result shows that with respect to the LSDRP, the amount of signature verification time in overall reprogramming time is very less, so the lightweight SDRP is more efficient than the improved SDRP using identity-based short signature scheme.


Result

Execution Time for Each Phase of the Improved SDRP and the Lightweight SDRP

	Improved SDRP using Short IBS Scheme	Lightweight SDRP
Key setup	121ms	104ms
User public/private key generation	120ms	82ms
User signing	242.116789019721ms	198.572649783256ms
Signature verification	2.9077504886816s	1.74051887297096s

Fig. 5 Execution time for each phase of the improved SDRP and the lightweight SDRP.

Fig. 6 shows the time in seconds to disseminate the entire program image to all nodes. As the packet loss increases, the improved SDRP performs worse than the lightweight SDRP.

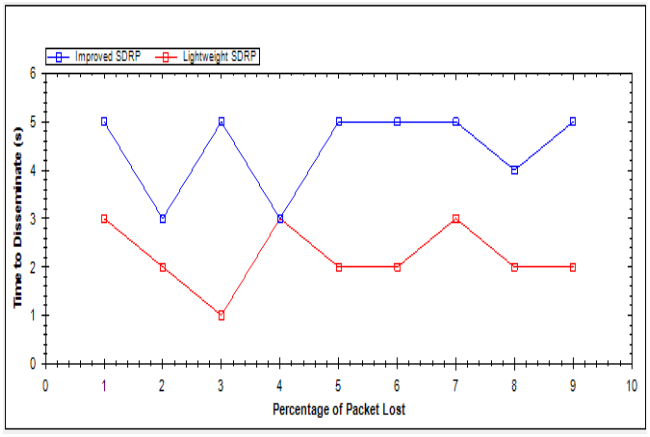


Fig. 6. Comparison of average dissemination time of a program image with increasing packet loss for each protocol.

Fig. 7 shows the number of packets transmitted on the data plane at different network densities. At different values of the network density, the amount of data packets transmitted by the lightweight SDRP, increases slowly, while the amount transmitted by the improved SDRP increases rapidly.

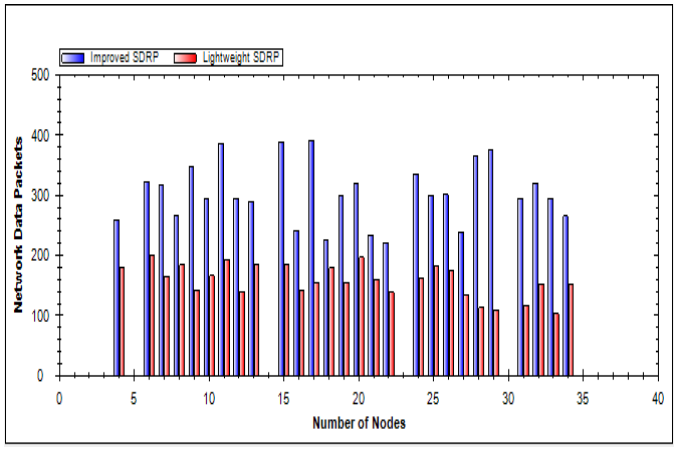


Fig. 7 Average number of packet transmitted on the data plane as a function of the network density, for each protocol.

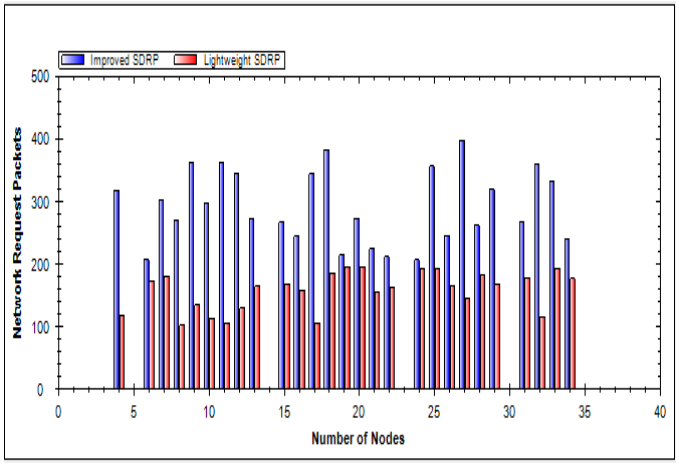


Fig. 8. Average number of packet transmitted on the control plane as a function of the network density, for each protocol.

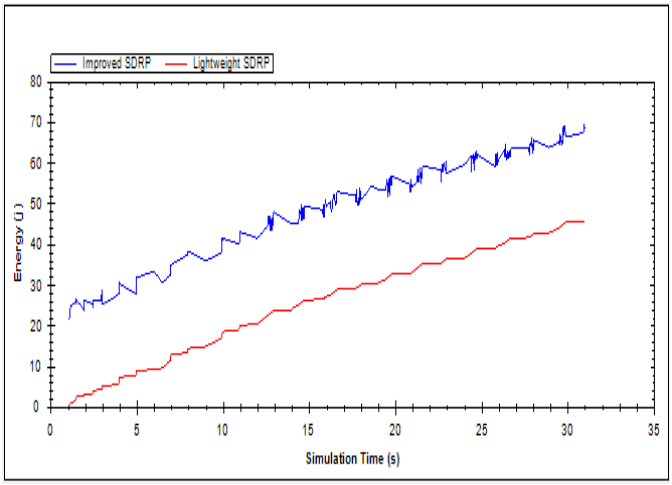


Fig. 9 Average energy use per node with increasing simulation time.

The number of packets transmitted on control plane is shown in the Fig. 8. At different values of the network density, the amount of request packets transmitted by the lightweight SDRP, increases slowly, while the amount transmitted by the improved SDRP increases rapidly.

Fig. 9 shows that the average energy consumed in joule (j) per node at different simulation time. As the simulation time increases, the lightweight SDRP consumes less energy

as compared to the improved SDRP. Thus, utilization of less energy by the lightweight SDRP extends the network lifetime.

VI. CONCLUSION

The improved SDRP has been successfully integrated with the rateless Deluge by using random linear codes. The improved SDRP using identity-based short signature scheme has been compared with the lightweight SDRP using metrics such as execution time, latency, communication on the data plane and the control plane, and network lifetime. Simulation result shows that the lightweight SDRP requires less execution time for each phase as compared to the improved SDRP using short IBS scheme. Moreover, compared to the improved SDRP, the lightweight SDRP minimizes latency at reasonable levels of packet loss, minimizes communication on the data plane and the control plane, and consumes less energy. Finally, it has been concluded that the lightweight SDRP is significantly more efficient in terms of delay, communication and energy than the improved SDRP using identity-based short signature scheme.

REFERENCES

[1] V. C. Gungor and G. P. Hancke, "Industrial wireless sensor networks: Challenges, design principles, and technical approaches," *IEEE Trans. Ind. Electron.*, vol. 56, no. 10, pp. 4258–4265, Oct. 2009.

[2] V. C. Gungor, B. Lu, and G. P. Hancke, "Opportunities and challenges of wireless sensor networks in smart grid," *IEEE Trans. Ind. Electron.*, vol. 57, no. 10, pp. 3557–3564, Oct. 2010.

[3] X. Cao, J. Chen, Y. Xiao, and Y. Sun, "Building-environment control with wireless sensor and actuator networks: Centralized versus distributed," *IEEE Trans. Ind. Electron.*, vol. 57, no. 11, pp. 3596–3604, Nov. 2010.

[4] D. He, C. Chen, S. Chan, and J. Bu, "SDRP: A secure and distributed reprogramming protocol for wireless sensor networks," *IEEE Trans. Ind. Electron.*, vol. 59, no. 11, pp. 4155–4163, Nov. 2012.

[5] S. Hyun, P. Ning, A. Liu, and W. Du, Seluge: Secure and DoS-resistant code dissemination in wireless sensor networks, *Proc. IPSN*, 2008, 445–456.

[6] A. Chlipala, J. Hui, and G. Tolle. Deluge: Data dissemination for network reprogramming at scale. Class Project, http://www.cs.berkeley.edu/˜jwhui/research/deluge/cs262/cs262a-report.pdf, 2003.

[7] J. W. Hui and D. Culler, "The dynamic behavior of a data dissemina - tion protocol for network programming at scale," in *Proc. SenSys*, 2004, pp. 81–94.

[8] D. He, C. Chen, S. Chan, and J. Bu, L. T. Yang, "Security analysis and improvement of a secure and distributed reprogramming protocol for wireless sensor networks," *IEEE Trans. Ind. Electron.*, vol. 60, no. 11, pp. 5348–5354, Nov. 2013.

[9] H. Du, Q. Wen, "An Efficient Identity-based Short Signature Scheme from Bilinear Pairings", in *Proc. IEEE CIS*, 2007, pp. 725-729.

[10] P. Motwani, P. Fulare, "Secure and Distributed Reprogramming Protocol for Wireless Sensor Networks using Identity-Based Short Signature Scheme," in *Proc. ICIAC*, 2014, pp. 29-33.

[11] P. Motwani, P. Fulare, "A Lightweight Secure and Distributed Reprogramming Protocol for Wireless Sensor Networks – A Review," in *Proc. ICAET*, 2014, pp. 18-22.

[12] P. Barreto, B. Libert, N. McCullagh,and J.-J. Quisquater, "Efficient and provably-secure identity-based signatures and signcryption from bilinear maps," in *Proc. ASIACRYPT*, 2005, pp. 515–532.

[13] N. P. Smart, "An identity based authenticated key agreemenr protocol based on weil pairing," *Electronic Letters*, vol. 38, no. 13, pp. 630-632, 2002.

[14] J. C. Cha and J. H. Cheon, "An identity-based signature from gap Diffie-Hellman groups," *PKC 2003,* LNCS 2567, Springer-Verlag, 2003, pp. 18-30.

[15] X. G Cheng, J. M Liu, and X. M Wang, "Identity-based aggregate and verifiably encrypted signatures from bilinear pairing," *ICCSA 2005,* LNCS 3483, 2005, pp.1046-1054.

[16] N. Koblitz, A. Meneze, "Pairing-based cryptography at high security levels," *Cryptography and Coding: 10th IMA International Conference*, LNCS 3796, Springer-Verlag, 2005, pp. 13-36.

[17] A. Hagedorn, D. Starobinski, and A. Trachtenberg, "Rateless Deluge: Over-the-air programming of wireless sensor networks using random linear codes," in *Proc. ACM/IEEE IPSN*, 2008, pp. 457–466.

[18] R. Merkle, "Protocols for public key cryptosystems," in *Proc. IEEE S&P*, 1980, pp. 122–133.