Security Analysis and Improvement of SDRP Using Diffie Hellman Algorithm in Wireless Sensor Network

Kavita J. Patil, Prof. Prashant Jawalkar

***Abstract*— Remote remaking in a remote sensor zone (WSN) is the path toward sending one more code picture or commands to sensor centers. Although all current dubious/secure recreating traditions rely on upon the concentrated approach, it is basic to support coursed rehashing in which variously endorsed framework customers can in the meantime and especially sensor centers are examined again without the inclusion of base station. Recently, a new secure and scattered recreating tradition known as SDRP has been introduced. Regardless, this paper recognizes a natural arrangement a gap in the customer pre-processing time of SDRP and shows that it is susceptible towards an emulate attack than can allow the enemy to extend mirror any endorsed customer to finish re-examining. This paper proposes a clear acclimation for settlement of the perceived security issue without compromising on any components of SDRP.Our objective to moreover upgrade the security and adequacy of SDRP using Diffie-Hellman.**

Index Terms*— Diffie-Hellman, Key exchange, reprogramming, security, sensor networks, user privilege

I. INTRODUCTION

Remote remaking is the route used to spread different code images or critical summons to sensor nodes through remote associations after a "remote sensor network" (WSN) is sent. Due to requirements of bug clearance and the inclusion of new functionality, reevaluating is an indispensable action of WSN. Most of the time WSN is sent to determine circumstances, example, the battle region, an enemy may mishandle the remaking tool to dispatch distinctive strikes. Along these lines, secure composition PC projects are and will continue being an imperative topic of investigated.

A lot of research has been conducted that focuses on secure rehashing, and recently numerous captivating traditions have been suggested. Regardless, all of them rely on the bound together approach that expects the nearness

of a base station, which has the master to remake sensor centers, as reflected in Fig. 1 Shockingly this approach is not tried, and true under the condition base station misses the mark or sensor center points lose the relationship with the base station, then the recreation is hard. What's more, some of the WSNs do not have a base station by any methods, and therefore, the joined approach is not related. Moreover, the brought together approach is not efficient is less versatile, and susceptible towards many potential ambushes in the long correspondence way. As of late, He et al. has presented a safe and dispersed remaking tradition known as SDRP that is novel work in Fig.1.

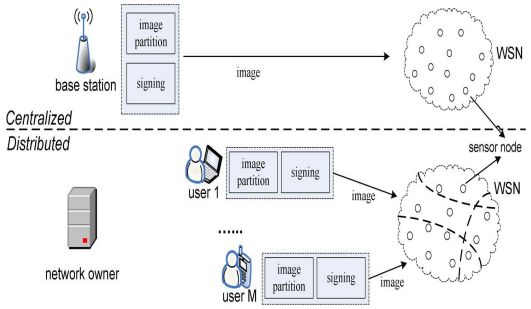


Fig. 1. System overview of centralized and distributed reprogramming approaches.

Centralized v/s Distributed reprogramming approaches its field. As a novel character, related check plan is being used in delivering open/private key join of each endorsed customer, SDRP is successful for resource compelled sensor center points and PDAs to the extent correspondence and limits necessities. Also, SDRP can fulfill all requirements of passed on re-evaluating , while keeping the advantages of the exceptional frameworks, for instance, Deluge [4] and Seluge [2]. Similarly, SDRP is being completed in an arrangement of advantage compelled sensor centres to display its profitability for all intents and purposes. In any case, this papers presents that a diagram issue remains in the customer pre-taking care of time of SDRP, and a foe can without quite a bit of an extend mimic any endorsed customer to do re-evaluating. To take out the recognized security feebleness, we propose a direct change on SDRP without losing any segments, (for instance, passed on re-evaluating, supporting unmistakable customer benefits, dynamic participation, flexibility, high capability, and generous security) of the main tradition. Likewise, we exhibit that, for security and adequacy thought, any beneficial character based stamp estimation which has survived various circumstances of open examination can be particularly used in SDRP. Furthermore, the paper r eports the trial outcomes of the upgraded SDRP in convenient workstation PCs and resource obliged sensor center points, which eventually show its efficiency. Wireless Sensor Network is a group of wireless nodes designed for the continuous sensing of information at human inaccessible locates. reprogramming is the monitoring conditions vary according to the environmental changes or other user requirements. Insecure transmission of reprogramming code to such areas can damage the entire operation of the network. To avoid this, Secure and Distributive Reprogramming Protocol (SDRP) proposed for user privilege maintenance. In this paper, Diffie-Hellman key (DH) exchange mechanism is implemented as an

Manuscript received April 19, 2017
Kavita J. Patil, Department of Computer Engineering, Department of Computer Engineering, BSIOTR(Womens), Wagholi, Pune- 41
Prof. Prashant Jawalkar, Department of Computer Engineering, Department of Computer Engineering, BSIOTR(Womens), Wagholi, Pune-41

improvement to the existing method to further improve security between the forwarding nodes.

II. REVIEW OF LITERATURE

Several different reprogramming protocols have been developed

A. Deluge Protocol

Deluge[4] is a reliable data dissemination protocol for propagating large amounts of data from one or more source nodes to all other nodes over a multihop, wireless sensor network. Deluge is the state of the art and included in TinyOS distributions. Deluge emphasizes the use of spatial multiplexing to allow for parallel transfers of data. However, since the design of Deluge did not take security into consideration .This approach, is vulnerable to Denial of Service (DoS) attacks.

B. Seluge Protocol

Secure and DoS-Resistant Code Dissemination in wireless sensor networks is a secure extension to Deluge, an open source, state of-the-art code changing system for wireless sensor networks. It provides security protections for code updating, including the integrity protection of code images and prevent from the attacks. Seluge properly authenticates advertisement and SNACK packets. Seluge uses a signature to self-sustaining process the authentication of a new code image. It can be efficiently verified by a regular sensor node, but it
takes a computationally powerful attacker a substantial amount of time to build a weak authenticator. Moreover, it cannot be pre-computed. Thus, this weak authentication mechanism provides an effective filter of forged signatures. As a result, Seluge is not subject to the same DoS attacks against signature verification[2].

C. Secure and Distributed Reprogramming Protocol

The SDRP[3] involves three phases: structure statement, customer pre-get ready, and sensor center point check. In the structure statement organize, the framework proprietor makes its open and private keys and a short time later chooses the rethinking advantage and the contrasting private key with the affirmed user(s). Simply individuals as one of the rules of the parameter are stacked on each sensor center point prior to association. During customer pre-taking care stage, framework customer enters the WSN and if he possesses another coded picture, it ought to build up the rehashing packages and then send them to the sensor center points. In the sensor center
point affirmation arrange, if the package check passes, then the centers recognize the coded picture.

D. Improved Secure and Distributed Reprogramming Protocol

SDRP and demonstrate that it is vulnerable to an impersonation attack by which an adversary can easily impersonate any authorized user to carry out reprogramming. In this paper fix the identified security problem without losing any features of SDRP. Moreover, in order to further improve the security and efficiency of SDRP, here established the identity based signature algorithm by Barreto can be directly employed in SDRP. Based on implementation results, demonstrate efficiency improvement over the original SDRP due to the following two reasons. First, its signature verification operation only needs one pairing computation and, hence, is among the most efficient ones. Second, the length of its signature is reduced due to bilinear pairing[1].

III. SYSTEM ARCHITECTURE

Diffie-Hellman key sharing algorithm is proposed for public key sharing. Diffie-Hellman is a computation is utilised for the development of a typical puzzle between two get-togethers. Diffie-Hellman is most of the time utilised as a system that exchanges cryptography keys for symmetric encryption estimations like AES. The count is secure in light of the way that the estimations of an and b, that are needed for the derivation of s are not transmitted over the wire by any methods.

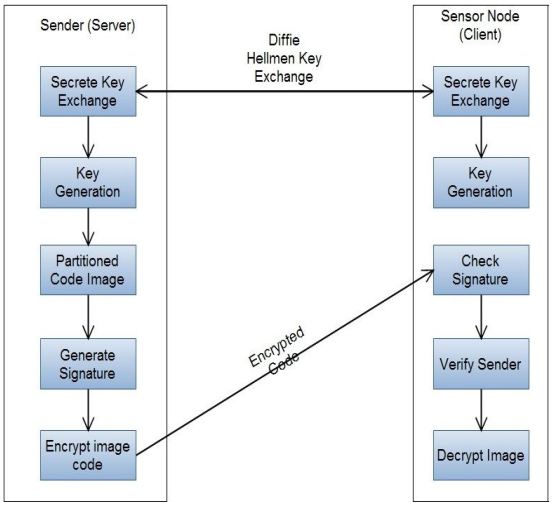


Fig. 2. System Architecture

As shown in Figure 1, Server node (Sender) and Sensor node (Client) will share the secrete keys using Diffie-Hellman key exchange algorithm which will generate the same key at both end for encryption and decryption purpose. After key generation at server, server partitioned the code image and calculates hash code for each packet. Hash value of each packet is added to the corresponding packet. To authenticate and integrity of the code image signature is generated. This signature is send to the client alongside the code image. On the client side, the key is generated after exchanging the secrete key. After the image code arrival, this calculates the signature of message. This signature is use for authenticate the sender. If the sender is valid the privileges are checked and the image code is decrypted. Using the hash value of the corresponding packet the actual image code is retrieved. If the entire authentication is approved, then only client accepts the image code.

1) System Initialization Phase

Consider G to be a cyclic additive group and G_T be a cyclic multiplicative group of the same prime order q.
Let P be a generator of G. Let $e\hat{}$: G × G → G_T be a bilinear map.
Choose two secure cryptographic has functions H_1 and H_2,where
H_1 : {0,1}*→ G and H_2: {0,1}*→ Z^*_q .

For a user U_j with identity $UID_j \in \{0,1\}^*$, the network owner sets modulus c $\in Z^*_q$, base d(primitive root modulo c). The public parameters {G, G_T , $e^{\wedge}$,q,c, H_1,H_2} are loaded in each sensor node before deployment.

***Private Key generation*:**

User and sensor node agree to use a modulus c and base d.
User chooses a secret integer a and compute A = d^a mod c, Then sends A to the sensor node.
Sensor node chooses a secret integer b and compute B= d^b mod c, Then sends B to the user.
User computes n=B^a mod c
Sensor node computes n= A^b mod c.
User and sensor node now share a same secrete key n.

B. User Pre-processing Phase

User U_j takes the following actions.
U_j partitions the code image to Y fixed-size pages denoted as page 1 through page Y . U_j splits page $i(1 \leq i \leq Y)$ into N fixed-size packets, denoted as $Pkt_{i,1}$ through $Pkt_{i,N}$. The hash value of each packet in page Y is appended to the corresponding packet in page $Y-1$. For example, the hash value of packet $Pkt_{Y,1}$ $h(Pkt_{Y,1})$ is included in packet $Pkt_Y-1_{,1}$. Here, $Pkt_{Y,1}$ presents the first packet of page Y . Similarly, the hash value of each packet in page $Y-1$ is included in the corresponding packet in page $Y-2$. This process continues until U_j finishes hashing all the packets in page 2 and including their hash values in the corresponding packets in page 1. Then, a Merkle hash tree [23] is used to facilitate the authentication of the hash values of the packets in page 1. We refer to the packets related to this Merkle hash tree collectively as page 0. The root of the Merkle hash tree, the metadata about the code image (e.g., version number, targeted node identity set, and code image size), and a signature over all of them are included in a signature message. The detailed information can be referred to in [17]. Lets assume the message m as the root of the Merkle hash tree and the metadata about the code image. Therefore, the authenticity and integrity of the new code image, U_j in order to create signature message undertakes below the action.
With the private key SK_j, U_j can compute the signature σ_j of the message m, where $\sigma_j = H_2(m) \cdot n$.
U_j transmits to the targeted nodes the signature message $\{UID_j,Pri_j,m,\sigma_j\}$, which serves as the notification of the new code image. SDRP relies on the underlying Deluge protocol to distribute packets for a given code image.

C. Sensor Node Verification Phase

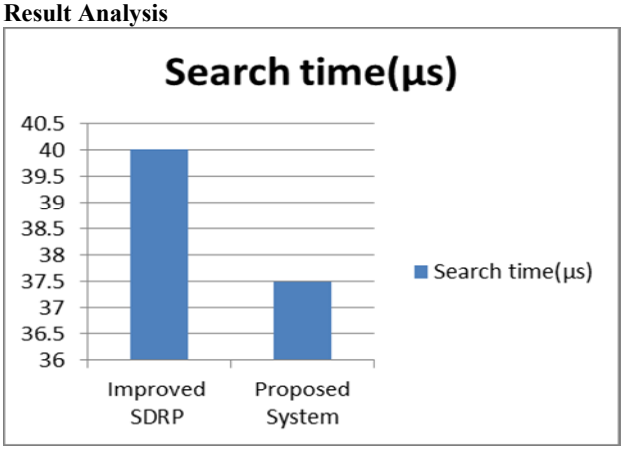
After each sensor receives signature message $\{UID_j,Pri_j,m,\sigma_j\}$, following verification is done by each sensor node verifies it as follows.
The sensor node first pays attention to the legality of the programming privilege Pri_j and the message m. After confirming the validity, the verification procedure takes next step.
Sensor node performs the following verification. Sensor node calculates the signature $\sigma_{j=} H_2(m) \cdot n$. This signature matches with the user signature, it is valid, and then goes to next step.

If the above-mentioned verification goes through, the sensor node considers that the message m and the privilege Pri_j are an authorised user with identity UID_j. So, the sensor node accepts the root of the Merkle hash tree constructed for page 0. Thus, the nodes can authenticate the h ash 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 on page 1 based on the one-way property of hash functions. In a similar manner, after the data packets in page i gets verified, a sensor node can easily authenticate the data packets in page i+ 1, where $i = 1,2,...,Y - 1$.Once all verification procedures mentioned earlier get passed then, only the sensor node accepts the code image.

Mathematical model

Let S, be a system such that,
S = {s, e,X,Y,T, fme,DD,NDD} where,
S- Main System model
s- Initial state at T< init > -SystemInitialization().
e- End state - SensorNodeVerification().
X- Input of System - Code Image ,Encryption Key, secret key
Y- Output of System – Code Image .
T- Set of serialized steps to be performed. SystemInitialization ().UserPreprocessing(),SensorNodeVerification().
Y - Encryption algorithm, Diffie Hellman-key exchange algorithm, hash value generation.

Result Analysis



Above graph shows that time required for search sensor node in proposed system is less than time required in the existing system.

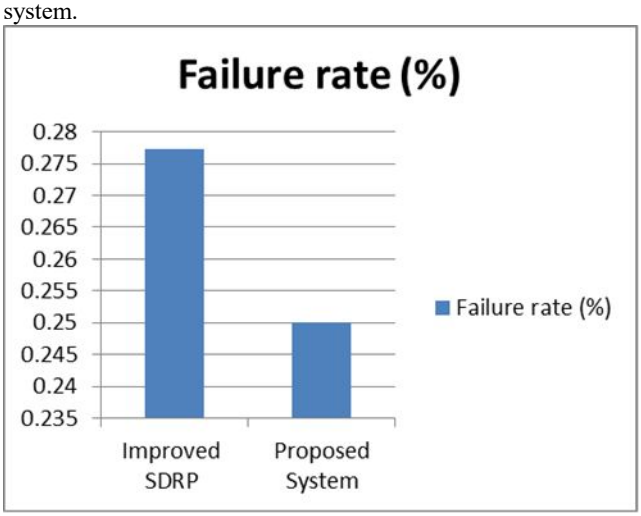
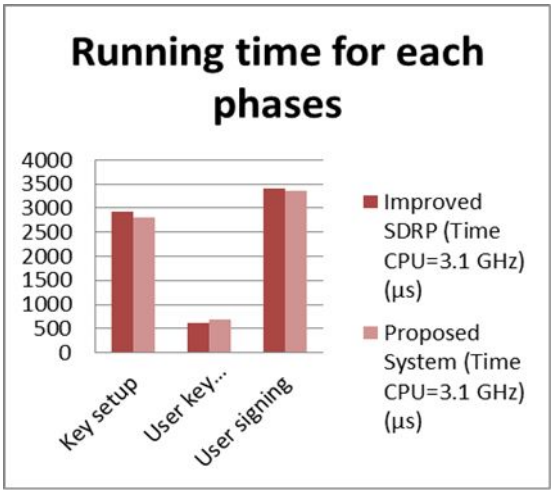
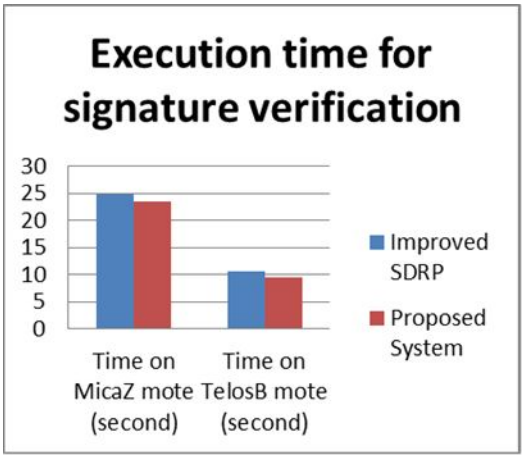


Fig. explains packet loss comparison. Packet loss occurs when one or more packets of data travelling across a network fail to reach their

destination.



Above graph shows that time required for running proposed system is less than time required for the existing system.



Above graph shows that time required for signature verification in proposed system is less than time required for the existing system.

CONCLUSION

In this paper, an inherent design weakness in the user preprocessing phase of SDRP is estimated. So Diffie Hellman algorithm is proposed, to provide the more efficient transmission. This algorithm keeps secrecy of sender and receivers reprogramming packets. Both sender and receiver share their public key and if both get identical values then they will send the private key. Hence it provides secured transmission. The Diffie-Hellman algorithm works perfectly to generate cryptographic keys which are used to encrypt the data being communicated over a public channel.

ACKNOWLEDGMENT

The authors would like to thank the researchers as well as publishers for making their resources available and teachers for their guidance. We also thank the college authority for providing the required infrastructure and support. Finally, we would like to extend a heartfelt gratitude to friends and family members.

REFERENCES

[1] Daojing He, Chun Chen, Sammy Chan, Jiajun Bu, and Laurence T.Yang, "Security Analysis and Improvement of a Secure and Distributed Reprogramming Protocol for Wireless Sensor Networks" IEEE Trans. Ind.Electronics, VOL. 60, NO. 11, NOVEMBER 2013

[2] S. Hyun, P. Ning, A. Liu, and W. Du, "Seluge: Secure and DoS-resistant code dissemination in wireless sensor networks," in Proc. IPSN, 2008,pp. 445456.

[3] D. He, C. Chen, S. Chan, and J. Bu, "SDRP: A secure and efficient reprogramming protocol for wireless sensor networks,"IEEE Trans. Ind. Electron., vol. 59, no. 11, pp. 41554163, Nov. 2012.

[4] J. W. Hui and D. Culler, "The dynamic behavior of a data dissemination protocol for network programming at scale,"in Proc. SenSys, 2004, pp. 8194.

[5] TinyOS, "An open-source OS for the networked sensor regime,"2012. [Online]. Available: http://www.tinyos.net/

[6] N. Bui, O. Ugus, M. Dissegna, M. Rossi, and M. Zorzi, An integrated system for secure code distribution in wireless sensor networks, in Proc. PERCOM, 2010, pp. 575581.

[7] Geoss, 2011. [Online]. Available: http://www.epa.gov/geoss/

[8] NOPP, 2012. [Online]. Available: http://www.nopp.org/

[9] R. Merkle, Protocols for public key cryptosystems, in Proc. IEEE Secur. Privacy, 1980, pp. 122134.

[10] K. Lin and P. Levis, Data discovery and dissemination with DIP, in Proc. IPSN, 2008, pp. 433444.