A Novel Key Management Protocol for Wireless Sensor Networks Based on PUFs

Ramin Bahrampour and Reza Ebrahimi Atani

Department of Computer Engineering, University of Guilan, Rasht, Iran
bahrampour.ramin@gmail.com, rebrahimi@guilan.ac.ir

Abstract

In this paper we focus on physical attacks on sensor nodes in a wireless sensor networks. Node capture attacks is one of the most dangerous attacks applied to WSNs which aims to capture a node in the network and try to steal some secret information. To be more specific we first survey node capture attacks and then a novel security mechanism to deal with these types of attacks will be presented. For this purpose a key management protocol is proposed based on authentication and data encryption preventing from node capture attacks. The key of protocol is generated by embedded PUF on sensor nodes chip. This key is unique and all tasks in protocol performed by this key and its products. The proposed solution is resistant against node capture attacks and is efficient in points of the cost and memory size.

Keywords: *Wireless Sensor Network, Node Capture, PUFs, Key Management protocol.*

1. Introduction

Wireless sensor networks (WSNs) consists of spatially distributed autonomous sensor nodes to monitor physical or environmental conditions because of that they have widespread applications in human communities and existence world. Necessarily location of sensor nodes is not determined and this feature makes it possible that we can get them in dangerous and inaccessible places [1, 2]. The sensor nodes due to constraints in size and cost have limitation in the amount of energy source, memory, computational speed and bandwidth. This resource limitations, creates risks and problems for WSNs that one of which is the lack of security in these networks. Another weakness of these networks is their layered architecture that makes the network more vulnerable to various attacks [3]. In many applications, these networks are used in environments hostile and inaccessible. Due to cost constraints, the use of secure hardware is not possible for all the nodes. So despite the conditions and limitations attacker can physically have access to nodes and takes the control of nodes and after that finds a way to control the full network. These attacks are the most common in physical layer of wireless sensor networks which are called Node Capture Attacks [4]. In this attack, the attacker takes control of the target node directly and by the attack, attacker can gets encryption key, the data collected or received information from other nodes. Also an attacker can eliminate the captured node or even attacker can convert it to a malicious node and try to make some other spy operations. In some other attacks the attacker can be replace a new node using the data gathered from the captured node [5].

In this paper, we describe the setup of node capture attacks and their risks in detail and also we provide a novel key management protocol based on the PUF key generation to deal with the attacks. The purpose of key management is to provide the key to establish secure communication between nodes dynamically. Key management protocol is responsible for production, distribution, storage, transport and disposal the key [6].

In this protocol in order to generate the key we use PUFs because this Physically Unclonable Functions can be implements easily in sensor nodes chip and able to create a unique keys. These type of functions cannot be destroyed by tampering therefore produces a high level of physical security. In this protocol, the required key generated by PUFs and several mathematical relations and these keys distributed in two different phases, before and after the establishment of the network [7].

If we provide a solution to be able to authenticate sender node in receiver section and data have been encrypted then the attacker cannot apply node capture attack to access information or perform subversion. Therefore in this paper, nodes by PUF keys and encryption methods performed authentication and encryption techniques without any restrictions. In this design, network structure determined hierarchical that asymmetric encryption uses for communication between nodes and to communicate between nodes and server symmetric encryption is used.

The remainder of the paper is organized as follows. In Section 2, security requirement in WSNs are discussed. Section 3 discusses the node capture attack in WSNs. Section 4 presents PUF that generates requirement keys in key management protocol. Section 5 provides the proposed key management protocol in different parts. Finally, Section 6 concludes from provided protocol deal node capture attacks.

2. Wireless Sensor Network Security

Due to the nature of the nodes in a WSN and hardware limitations and wireless communications of sensor networks, they are vulnerable to attack invaders, thus security protection of these networks is very hard and important [3]. There are two categories of security requirements to maintain security in WSNs, a public security that includes Confidentiality, Integrity, Authentication and Availability (CIAA) and other group are unique security that such as Data Freshness, Self Organization, Time Synchronization and Secure Localization [5, 8]. It is necessary that solution provided must be based on security requirements.

3. Node Capture Attack

Node capture attacks are happened on physical layer, but impacts on all other layers and are considered as an active attack. In this attack, the attacker takes control of the target node directly. By this attack, the attacker can achieve the important information such as encryption key, data collected and received information from other nodes. Other actions that an attacker can do on the captured node are destruction or change to a malicious node to performing subversion operation later [9, 10].

4. PUF

Physical unclonable functions are new creative functions in order to extract codes with complex intrinsic properties of its integrated circuits instead of storage them in digital memory. From the perspective of security, unclonable and unique properties of PUF key are considered. These functions are destroyed deal physical attacks thus they creates a high security in generation volatile keys [7]. By embedding a PUF on a sensor node chip, makes possible it that nodes be identifiable and can’t be reproduced physical uniquely [11, 12, 13]. These functions generate keys in a challenge and response process. The inputs to a PUF are generally called challenges and the outputs are called responses [11]. An attacker to attack

needs to the exact simulation from PUF, but the simulation due to used nonlinearity delays circuits is too hard.

4.1. Generating Encryption Key by PUF

PUF circuits output is inappropriate as a key to encryption. Due to noise, the outputs are slightly different in each time (even on a processor and to a challenge similar). So should be added an error correction capability to produce same bits. Figure 1 shows how with two new functions to error correction increases circuit reliability. First function called PUF-Calibrate that challenge(C) takes as input and computes response(R) with syndrome(S) that returns as output [14].

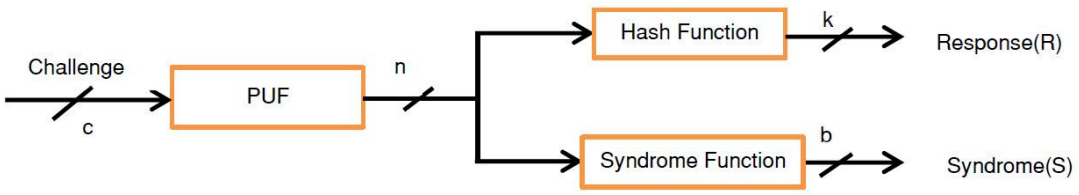

Figure 1. The generation of PUF key and syndrome [14]

The second function called PUF-Regenerate with two input C challenge and S Syndrome how regenerate PUF key is shown in Figure 2. Syndrome function corrects errors in the PUF output circuit delay before response be irregular [14].

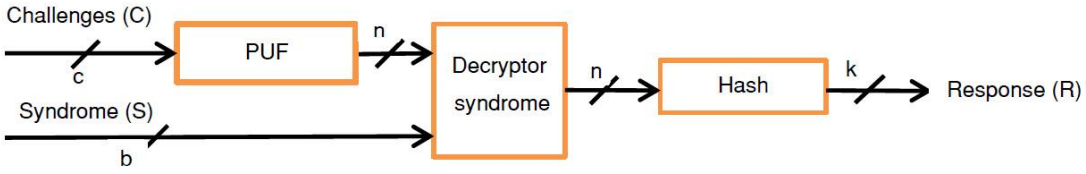

Figure 2. Reproducing the PUF key [14]

4.2. Arbiter PUF

Today, various PUFs are implemented to key generation on sensor nodes and in this topic have been done much assessments and studies [11, 12, 15]. Mainly used from the SRAM PUF and Arbiter PUF in various articles because they have low energy consumption and benefits from hardware small, simple and fast [15, 16]. In recent years the designed a Sensor PUF but due to get affected from environment temperature and its interaction on output, its reproductive capability this PUF is low [17]. As mentioned to generate a PUF key, it is required that PUF circuits be implemented on sensor nodes chips. The PUF selection should be compatible with the sensor nodes and WSN limitations. These limitations include weakness energy source, low-power processor, limited memory and small hardware chip [12]. Therefore in this paper is used an Arbiter PUF. This type of PUF is a small, fast and easy to implementing on sensor nodes chip. The other hand due to have a small hardware circuit doesn't need a powerful processor and large memory and can operate with low power consumption. Also this PUF against change temperature affected less and this feature makes reproducible better [15].

5. Key Management

Key management is based secure communication among sensor nodes by cryptographic algorithms. Purpose of key management is providing key for secure communication between nodes dynamically. The key management is responsible generation, distribution, storage, transport and disposal of the key [18, 19, 20]. The most important criterion in key management issue in sensor networks is energy consumption of course sensor nodes have other limitations such as less memory and low power computing which makes management keys very difficult [6, 21].

In this proposed protocol to encryption and authentication uses from two encryption methods symmetric and asymmetric encryption. Symmetric encryption performs faster and consumes less energy than asymmetric encryption symmetric encryption [21]. For this reason to communications between node and server uses symmetric encryption and because for communication between nodes using from it is impossible, asymmetric encryption is benefited. In this paper, key management protocols generally divided into the following four categories:

- Generation and distribution key in network
- authentication
- Storage data on the sensor node
- Secure data transmission

5.1. Generation keys by PUF and distribute in network

As mentioned PUF circuits are embedded on sensor nodes chip to generate unique keys. In this protocol, the key generation and distribution are performed in two phases, before and after the network establishment.

5.1.1. Generation and storage key before network establishment: In the proposed protocol before deployment of network, two pair keys are assigned to each sensor node:

1. Public and Private keys (P,V)

2. Challenge and Response keys (P,R)

Public and private pair key (P,V) to encryption messages and challenge-response pair key(P,R) for authentication as well as to generation (P,V) are used. As seen in the above formula used public key (P) in both pair key and also are equal with each other. Like asymmetric key encryption, server assigns one pair public key and private key (P,V) to each node. The public key P is unique for each node. To generation PUF key uses P as challenge and in output generate R as response and S (the syndrome function). The unique pair key (P, V) and (P, R) for each sensor node obtains and stores on the server. But due to risks of physical attacks on sensor nodes, keys V and R don't stores on nodes memory. Here only P and S (syndrome key) stored on node to generate R and key V calculates with a linear relationship (M) between V and R. The access of attacker to this information doesn't create any danger for sensor networks. This information is saved on nonvolatile memory as EEPROM or Flash before distributing sensor nodes on the network. Also due to use asymmetric key cryptography in this scheme, a copy of the server's public key is stored on each node. Relation M(R,V) is a simple linear relation that calculated on the server before distribute nodes. In a simple math example, suppose server assigned (11,19) to node X as public and private pair key and node Applies P=11 as challenge to its PUF and for example

produced on the PUF output R=39. Thus challenge and response pair key of node is (11,39) then M relation is following and saves on X's memory :

$$R = 39\ ,\ V = 19 \longrightarrow \quad M \rightarrow V = 2R + 1 \tag{1}$$

Also to communicate on the network, (is described in the next section) a K value is requirement that it obtains from relationship M' between the pair keys challenge-response (P, R) and stored on sensor nodes memory and server too. Relation M' is a perfectly linear relationship and it saves on sensor node. This relation doesn't create any threat to the network. Also this relation is identical for all nodes and is stored before distributed sensor nodes in the environment.

$$K = M'(P, R) \tag{2}$$

Figure 3 shows how to generate and store keys before deployment of network on nodes and server.

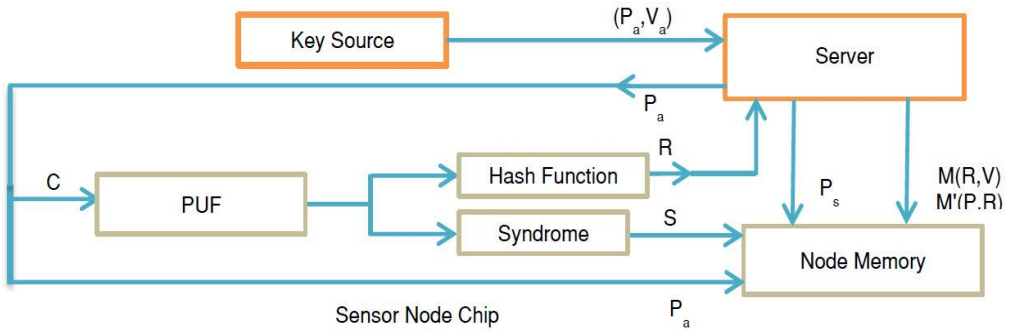

Figure 3. The Generation and storage of keys before deployment

But after the establishment of the network that is described in next sections, nodes to generate V, first produces R through PUF and obtains V by relation M and value R. The actions applies real time and none of the R and V keys don't stores in nodes memory and even after use these keys is removed from on the cache node. For this reason aggressive never can access encryption keys of nodes in the network.

5.1.2. Network structure and disturbing key after establishment network: Key management protocol must be compatible with the topology of the network. Therefore, in the design is used from a hierarchical structure that is shown in Figure 4. In this structure, network divided to several cluster and any cluster is formed from a cluster head and many sub nodes. The cluster heads have an additional supporter battery and also from point of view processor and memory are more power and better than the other sensor nodes [4]. After deployment of network, nodes selects a cluster head based on criteria of network as well as cluster heads are linked together in this structure.

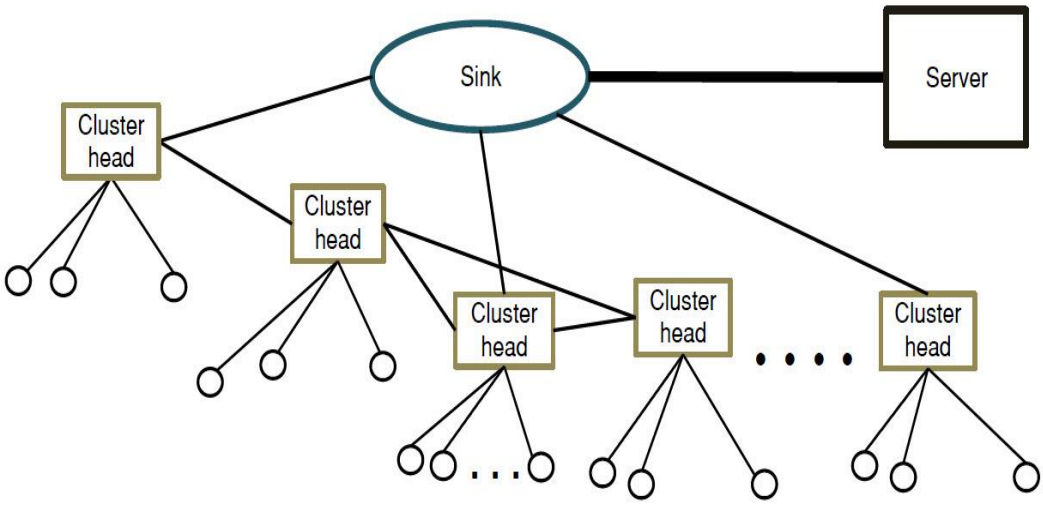

Figure 4. The Network Structure

After distribute nodes in environment and deployment of the network perfectly, each node determines its cluster head. Then, the server sends the public key of the cluster heads (P_C) for all of their sub nodes and also sends public key of all sub nodes for their cluster heads. Then nodes after receive the public keys, stores them on its memory. Thus all the sub nodes have the public key of their cluster head as well as the public of sub nodes is stored in its cluster head's memory. To communication in network is required to authentication between the cluster head and its sub nodes exist. For this purpose, the server calculated relationship between K value's sub node and amount of PUF key(R) related to its cluster head with relation $M''_C(K,R_C{}^1)$ and finally sends it to the cluster head. In a simple example, with assuming K = 13 and PUF key of cluster head $R_C = 78$ then relation $M''_C(K_i,R_C)$ is calculated as follows:

$$R_C = 88,\ K_i = 13 \longrightarrow M''_C \rightarrow R_C = 6K_i + 10 \quad (3)$$

Additionally server calculates relation M''_C to communication between the cluster heads and sends to them. Also when cluster head sends message to node, the cluster head node have to be authenticate in the sensor nodes that for this purpose, the is used from relationship $M''_N(K_C,R_i)$. This relationship after deployment of the network, server computes relation between K of cluster head and R of them sub nodes and sends to sub nodes. About use from relations M''_C and M''_N is explained in next section.

Table 1 shows information related to one node which is stored in its memory, on its head cluster and server is observed in before and after distribution of the nodes.

[1] Cluster

Table 1. All key information of the network

	Information of sensor node on its memory	Information of all nodes on server	Information of cluster head on its memory
Before the network deployment	Public key (P) Syndrome key (S), Relation of M(R,V), Relation of M'(P,R), Server public key(P_S)	Public and private pair keys of all nodes (P,S), Challenge and response pair keys of all nodes (P,V), K for all nodes	Public key (P) Syndrome key (S), Relation of M(R,V), Relation of M'(P,R), Server public key(P)
After the network deployment	Public key of its cluster head(P_C) Relation of M"$_N$(K_C,R_i)		Public key of other cluster head, Public key of sub nodes, Relation of M"$_C$(K_i,R_C)

5.2. Authentication

Each sender in network to send data should be identified in destination. In this protocol, except that the sender of message should be verification of authenticity, also to more security, cluster head must authenticates in the server in different time slots. In the proposed protocol, generally authentication is in three different ways:

5.2.1. Node authentication in server: Authentication in the server is identical for sensor node and cluster head node and is shown in Figure 5. In order to first node loads its public key and syndrome key stored on memory and applies them to the embedded PUF on its and ultimately produces R in output. Challenge and response pair key obtained, encrypts in symmetric encryption method by the R key and sends to the server. After the server receives the message, it decrypts by the private key of the node. Then server compared public and private keys received with public and private keys stored on its memory related to the node and if the both keys were identical then node is confirmed else because it is possible that environmental factors have been impacted on the PUF output of node, server requests from node to again authentication. The node with receiving request renew authentication from server, like the previous times that mentioned above server prepares authentication packet and sends to the server. Once the server receive the authentication packet repeats identity confirmation operation. If keys weren't identical frequently then occurs two modes. If target node was a sensor node, by a message from server to cluster head, the node removes from network and if target node was a cluster head, it removes from network and the subdirectory sensor nodes transferred to other subdirectory of cluster heads. Then the server like previous times, M_C (K_i, R_C) and M_N (K_i, R_C) for the cluster head nodes and sensor nodes calculates and sends to them.

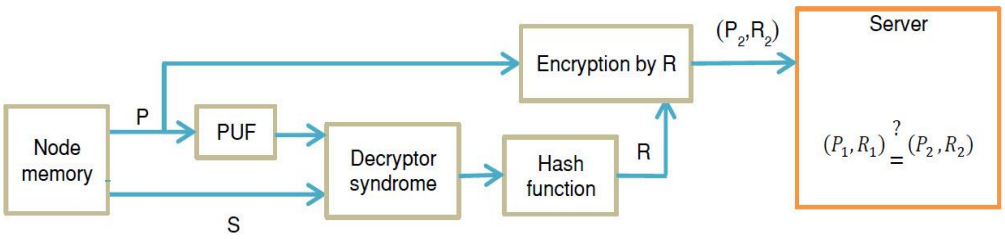

Figure 5. Node authentication in the server

As expressed except sender authentication before sending data, to increase level of network security it is required that performed authentication of cluster head in server in different time slots. The authentication could apply based on security criteria and network performance in different intervals. The criteria are such as: security level, network traffic and power of cluster head node. It is possible that based on standards of network sets amount of time slot.

5.2.2. Server authentication in node: In this protocol, server muse be authenticate in receiver node to send data. Thus first server encrypts K value by R of receiver node and sends for the node. Then the node receives a packet from the server and begins to decryption the message. First node applies challenge to the PUF embedded and generates response key(R). Then computes private key (V) with the R and M relationship. The node by V decrypts packet and with M'(P,R) relation must verify the authenticity of the sender that it be assured. With placing the required values of M', earns K. The node compares K calculated with K received from the server where the both value be equal, server is authenticity verified. But if not confirmed node authentication then must authenticate renew like way that the above mentioned.

5.2.3. Sensor node authentication in cluster head node and reverse mode: In the key management protocol, sensor nodes for data transmission to the cluster head should be authenticate in destination. Sensor node produces the R key to verify the authenticity in cluster head and then obtains K by the M'(P,R). This value is encrypted by the public key of cluster head, and sends for it. Once the cluster head receive packet, generates its V key by PUF output and its M relation. Now encrypts packet by its V and obtains K value of sensor node. In next step loads $M''_C(K,R)$ relation and by insert K of sender node and its R value in $M''_C(K,R)$ relation authenticates sensor node. The sensor node is verified where M''C relationship is correct else cluster head requests authentication again. This authentication operation performed anew and if not verified second authentication then both sensor node and cluster head must do identity verification in server. Due to maybe cluster head have been problem. The server with receiving the both authentication packet, checks them and removes the damaged node.

Cluster head authentication in sensor node methods is identical sensor node authentication in cluster head methods but uses M''_N relationship Instead M''_C relationship.

5.3. Data storing on nodes

In most applications of networks, nodes doesn't send data directly to the server or other nodes but rather in most cases the data stores on the sensor node nonvolatile memory and sends later. To protect the security of the data collected from the environment or from other nodes, they must be encrypted to storage on memory. Because the symmetric key cryptography is faster and less energy consumer than public key cryptography to cryptographic operations used from this method and by R key nodes.

5.4. Data transmission

One of the key management tasks, establish a secure connection to send and receive packets on the network. Generally sending data for the three components of the server, cluster head nodes, sensor nodes are divided into the following procedures:

5.4.1. Sending data from one node to the server: Sending data from a sensor node to the server is similar for sensor node and cluster head node. This method performs like authentication in server and just data adds to packet. In order to data and authentication encrypts by R key node with symmetric encryption method and sends to server. Server decrypts received data by R key of sender. If be confirm the identity of sender then server uses from data.

5.4.2. Sending data from server to nodes: Sending data from server to nodes is same for sensor node and cluster head node perfectly. First server loads K of receiver node to its authentication in destination and with data puts in packet. Then server encrypted packet by R key of receiver node and symmetric encryption technique. After the receive packet from server, node uses from its R to decryption it. In next step the node achieves K by M'(P,R) and compared with received K. If the relation be correct then identity of server is confirmed and server could use from data.

5.4.3. Sending data from one node to the neighbor node: Suppose according to Figure 6 node a wants to send data to node *b*.

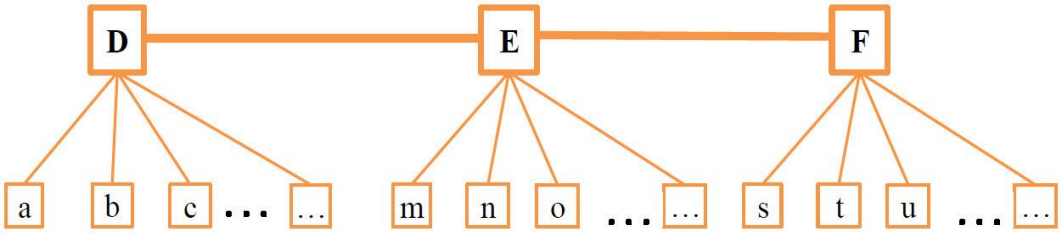

Figure 6. Communication between nodes and cluster heads.

As Figure 7 because node *a* can't authenticated in node *b*, first a should send data to cluster head *D* and data be send from cluster head *D* to node *b*. In this way, beginning node a applies its public key as challenge to PUF and generates value R_a. Then *a* loads relation M' from memory to generate K_a and calculates its *K*. In next step, node *a* puts data and K_a in packet and encrypts it by public key of cluster head *D*. Then sends to cluster head *D*.

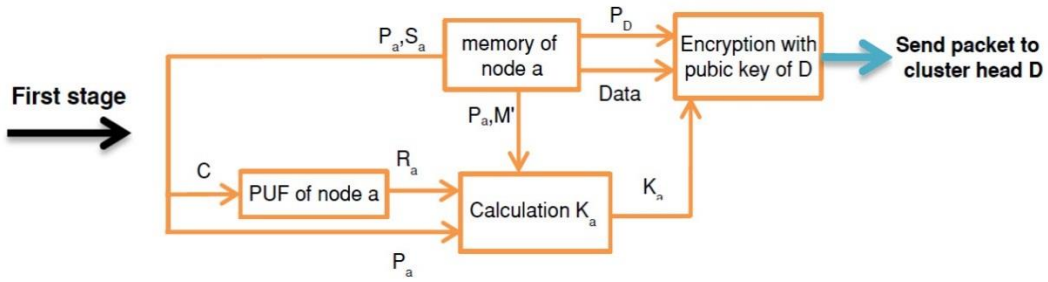

Figure 7. Send data from one node to neighbor node (first Stage)

In next stage, cluster head *D* receives packet from node *a*. As Figure 8 in first step, cluster head *D* loads challenge from its memory and applies it to PUF and generates response key R_D. Then by relation M and R_D computes value V_D and decrypts packet with it. Now node *a* must be authenticate in cluster head *D*. In order to *D* inserts values K_a and R_D in relation M''_C and if it was correct, node *a* is confirmed in *D*. In next step, cluster head *D* must send received data

and its authentication to node *b*. For this purpose, first *D* calculates K_D by relation M' and R_D and with data puts in packet. Then encrypts it by public key of node *b* and finally sends to node b.

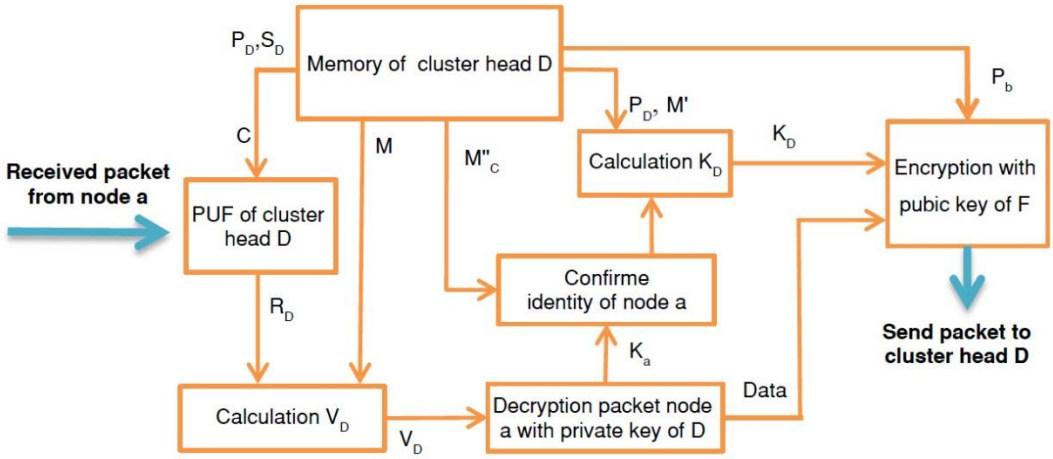

Figure 8. Send data from one node to neighbor nodes (Second stage)

In last stage, node *b* receives packet from cluster head *D* and must generates its private key. In order to, first node *b* applies its challenge to PUF and gets response key. In next step, obtains private key by R_b and relation M. Then node *b* decrypts packet by V_b and checks identity of cluster head *D* and if was confirmed, node *b* uses from sent message of node *a*. The last stage is shown in Figure 9.

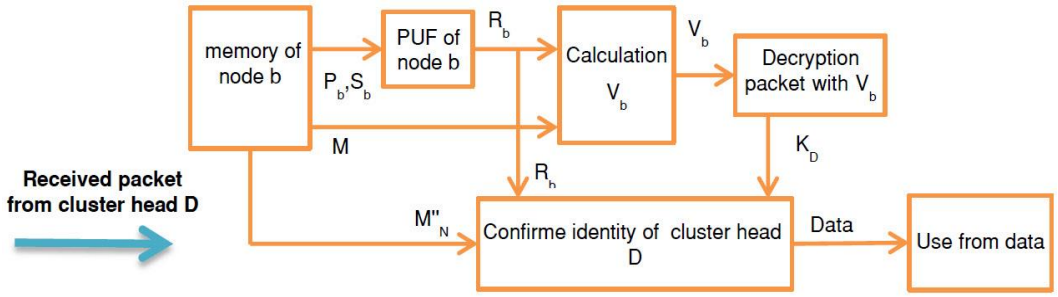

Figure 9. Send data from one node to neighbor node (third stage)

5.4.4. Sending data from one node to node of other cluster head: In this type, according to Figure 6 suppose node *b* Intends to send data to node *t*. For this data transmission, beginning *b* should encrypts K_b and data by P_D and send message to *D*. Node *D* once the receive message, decrypts it by V_D and authenticate *a* with relation $M''_C(K_a,R_D)$. Then *D* encrypts data and K_D by P_F again and forwards to *F* cluster head. Node *F* like previous step, performs decryption with V_F and applies identity confirmation by relation $M''_C(K_D,R_F)$. If be confirmed identity of *D* then encrypts data and its *K* by P_t in asymmetric encryption and sends to node *t*. Eventually node *t* receives message of node *b* from cluster head *F*. Now node *t* decrypts it and authenticate cluster head *F* by relation $M''_C(K_F,R_t)$. If Identity of *F* was confirmed then *t* uses from received data.

5.4.5. Sending data from one node to whole network (broadcast): Transfer data from *a* particular node to the whole network can be done in several ways. But the best and most cost effective method is this way that nodes sends data to the server and from the server be send to all nodes. Suppose node a wants to broadcast data in network. First the node must calculate value K_a by PUF key and relation M'. Then node encrypts K_a and data by response key (R_a) and sends to server. The server receives data and decrypts it by R_a and eventually checks node authentication. If was confirmed then server encrypts its authentication and data by nodes response keys and then sends for all nodes. Ultimately each node receives its message and decrypts it by its R key. Nodes after the confirm server authentication, uses from data.

6. Results

In this paper a novel key management protocol for WSN is proposed with is secure against node capture attacks. The security of protocol is based on the use of PUFs and their intrinsic properties. PUFs can be implemented during the manufacturing node chips and will have extra costs. PUF circuits by their unique physical characteristics are able to reproduce the key in any time or any place. Therefore without any additional cost and other limitations, With the implementation of PUF circuits on chips nodes, any attempt to remove PUF or each physical attack by the attacker, causes to change the key generation process thus changes the primary key and in no way don't allows to attacker to access information on the sensor node. As a result circuits of PUF remains safe against physical manipulation and capture of nodes attacks and increases network security level.

Key management protocols of symmetric encryption for sensor networks usually were in two forms which each of had their own limitations or problems. In the first form, was needed to larger memory for storing a large number of keys that it had additional costs. In the second form, to providing key must be used other nodes as mediated that in this method wastes energy source of nodes and increases network traffic. Additionally, none of methods can't create secure perfectly and the attacker can access to encryption keys [6]. But in the proposed protocol doesn't exist none of the mentioned limitations. Because the PUF embedded on the sensor node is able to produce its own unique key. Therefore don't need to the keys storage on node memory and the attacker is unable to access. Also the proposed protocol allows nodes to adding or reducing from the server side. Because in the protocol relationship of the generation and distributing keys between nodes and cluster head and server is defined in a way that doesn't creates any prohibition to increase or decrease the nodes in the network.

Invasive in this layer can only destroy a sensor node and for this purpose must access physically to each nodes. Considering to hierarchical structure used in the protocol, the attacker with destruction a cluster head node can create costs such as increase of network traffic and more energy consumption of nodes because after removing the cluster head, all its sub nodes must be transfer to another cluster heads and calculated for all of them $M''_N(K_C,R_i)$ and $M''_C(K_i,R_C)$ relations. Also due to the weak of the layered sensor networks duties, this protocol is limited to the physical layer security. However, the protocol can prevent some of the attacks on higher layers and reduces the damages of attacks.

By use data encryption, attackers aren't able to perform active attacks in higher layer and only can do inactive attacks such as removing packages or set up routing attacks.

7. Conclusion

In this paper by embedded PUF on node chip provided a solution to prevent from node capture attacks. PUF circuits by their unique physical characteristics are able to reproduce their key in any time and place. Here with embedded PUF on sensor nodes chip, nodes are

able to create a unique key without any limitations and additional costs. We showed generation key process with PUF and used it to authentication sender in destination and encryption key. In order to we designs a management key protocol that performs authentication and encryption operation by PUF key. This protocol have feature that allows to network for increase or decrease nodes and also the solution saves in memory size and energy consume in encryption techniques.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, "A survey on sensor networks", IEEE Communications Magazine, vol. 40, **(2002)**, pp. 102-114.
[2] J. Yick, B. Mukherjee and D. Ghosal, "Wireless sensor network survey", Computer Networks, vol. 52, **(2008)**, pp. 2292-2330.
[3] Y. Wang, G. Attebury and B. Ramamurthy, "a Survey of Security Issuse in Wireless Sensor Networks", IEEE Communications survey& Tutorials, the Electronic Magazine of Original Peer-Reviewed Surve Article, **(2006)**.
[4] C. F. Garcia-Hernandez, P. H. Ibarguengoytia-Gonzalez, J. Garcia-Hernández and J. A. Pérez-Diaz, "Wireless Sensor Networks and Applications: a Survey". International Journal of ComputerScience and Network Security, vol. 7, **(2007**), pp. 264-273.
[5] J. Sen, "A Survey on Wireless Sensor Network Security", International Journal of Communication Networks and Information Security, vol. 1, **(2009)**, pp. 55-78.
[6] J. Zhang and V. Varadharajan, "Wireless sensor network key management survey and taxonomy", Journal of Network and Computer Applications, vol. 33, **(2010)**, pp. 63-75.
[7] R. Maes "Physically Unclonable Functions: Constructions, Properties and Applications", Katholieke Universiteit Leuven, Doctor of Philosophy, **(2012)**.
[8] S. Mohammadi and H. Jadidoleslamy, "A Comparison of Physical Attaks on Wireless Sensor Networks". International Journal of Peer to Peer Networks (IJP2P), vol. 2, **(2011)** April, pp. 24-42.
[9] Z. Benenson, P. M. Cholewinski and F. C. Freiling, "Wireless Sensor Network Security", Edited J. Lopez, and J. Zou, IOS Press, **(2008)**.
[10] G. Padmavathi and D. Shanmugapriya, "A Survey of Attacks, Security Mechanisms and Challenges in Wireless Sensor Networks", International Journal of Computer Science and Information Security, vol. 4, **(2009)**.
[11] R. Maes and I. Verbauwhede, "A Discussion on the Properties of Physically Unclonable Functions", TRUST-2010 Workshop on Security Hardware in Berlin, Germany, **(2010)**.
[12] S. Meguerdichian and M. Potkonjak, "Security Primitives and Protocols for Ultra Low Power Sensor Systems", University of California, Los Angeles, **(2011)**.
[13] J. Guajardo, S. Kumar and P. Tuyls, "Key Distribution for Wireless Sensor Networks and Physical Unclonable Functions", Secure Component and System Identification Workshop - SECSI Berlin Germany, **(2008)**.
[14] M. Ayat, "Design Implementation of an Unclonable Crypto processor Based on PUFs", MSc Thesis, Iran University of Science and Technology, **(2011)**.
[15] M. Ayat, R. E. Atani and S. Mirzakuchaki, "On Design of PUF-Based Random Number Generators", International Journal of Network Security & Its Applications (IJNSA), vol. 3, **(2011)**.
[16] G. N. Selimis, M. Konijnenburg, M. Ashouei, J. Huisken, H. de Groot, V. van der Leest, G. J. Schrijen, M. van Hulst and P. Tuyls, "Evaluation of 90nm 6T-SRAM as Physical Unclonable Function for Secure Key Generation in Wireless Sensor Nodes", In IEEE International Symposium on Circuits and Systems (ISCAS), (2011), pp. 567-570.
[17] K. Rosenfeld, E. Gavas and R. Karri, "Sensor Physical Unclonable Functions", polytechnic Institue of NYU, Brooklyn, **(2010)**.
[18] Y. Xiao, V. K. Rayi, B. Sun, X. Du, F. Hu and M. Galloway, "A Survey of Key Management Schemes in Wireless sensor network" Computer Communications, vol. 30, (2007), pp. 2314-2341.
[19] Y. Jeong and S. Lee, "Secure Key Management Protocol in the Wireless Sensor Network", International Journal of Information Processing Systems, vol. 2, **(2006)**.
[20] T. Laskar and D. Jena, "A Survey on Key Management Issues in WSN" International Journal of Engineering and Innovative Technology (IJEIT), vol. 1, **(2012)**.
[21] F. Amin, A. H. Jahangir and H. Rasifard, "Analysis of Public-Key Cryptography for Wireless Sensor Networks Security", World Academy of Science, Engineering and Technology, vol. 41, **(2008)**, pp. 529-534.

Authors

Ramin Bahrampour

Ramin Bahrampour received his B.S. degree in applied software engineering from University of Chalus, Chalus, Iran, in 2008, and M.S. degree in information technology engineering from University of Guilan, Rasht, Iran, in 2013. He researched in image processing and image compression in year 2010. During the years 2011-2012, his research interests include cryptography, key management, attack simulation and prevention in wireless sensor networks and its security in general.

Reza Ebrahimi Atani (Corresponding Author)

Reza Ebrahimi Atani was born in 1980. He received the B.S. degree in electrical engineering from the University of Guilan in 2002 and the M.Sc and PhD degrees in electronics from Iran University of Science and Technology in 2004 and 2010 respectively. Since 2010, he is an assistant professor in Computer Engineering Department at the University of Guilan. His current research interests include Computer and Network Security, cryptographic hardware and embedded system (CHES), and design of VLSI circuits.