EFFICIENT GROUP AUTHENTICATION AND KEY EXCHANGE SCHEME FOR IOT ENVIRONMENT

[1]DAE-HWI LEE, [2]IM-YEONG LEE

[1,2]Department of Computer Science and Engineering, Soonchunhyang University, South Korea

E-mail: [1]leedh527@sch.ac.kr, [2]imylee@sch.ac.kr

ABSTRACT

User authentication verifies that a user is a legitimate user. Such security is essential. As the Internet of Things (IoT) continues to develop, user authentication has become critical. In an IoT environment, many interconnected devices share data. Such devices can form small or large groups and communicate with outside groups. With a group, a leader communicates with followers such as smart devices. If a group is large, the group leader incurs a very large operation overhead in terms of secure communication. When multiple devices request simultaneous communication, many sessions must be authenticated, and keys provided. Here, we propose an XOR-based group authentication technique enabling group leaders to perform authentication and key exchange very quickly, and a CRT-based group authentication technique that can be used when data are less distributed.

Keywords: *IoT, Group-based Communication, Group Authentication, AKE, Key Exchange.*

1. INTRODUCTION

The Internet of Things (IoT) exists within the Wireless Sensor Network (WSN) environment, allowing all objects to connect to the Internet to provide various device-based services. Many convenient services have already been developed, and the Wireless Body Area Network (WBAN) will expand services within larger environments, creating smart homes and cities.

In the IoT environment, many devices and users share data. Among devices participating in communication, gateway- and server-class devices have good computing power, but most sensor devices do not. Therefore, any security protocol (such as authentication) must be lightweight if it is to function well even with devices with low computing power.

In WBAN environments, data are collected via sensors or smart watches and transmitted to the

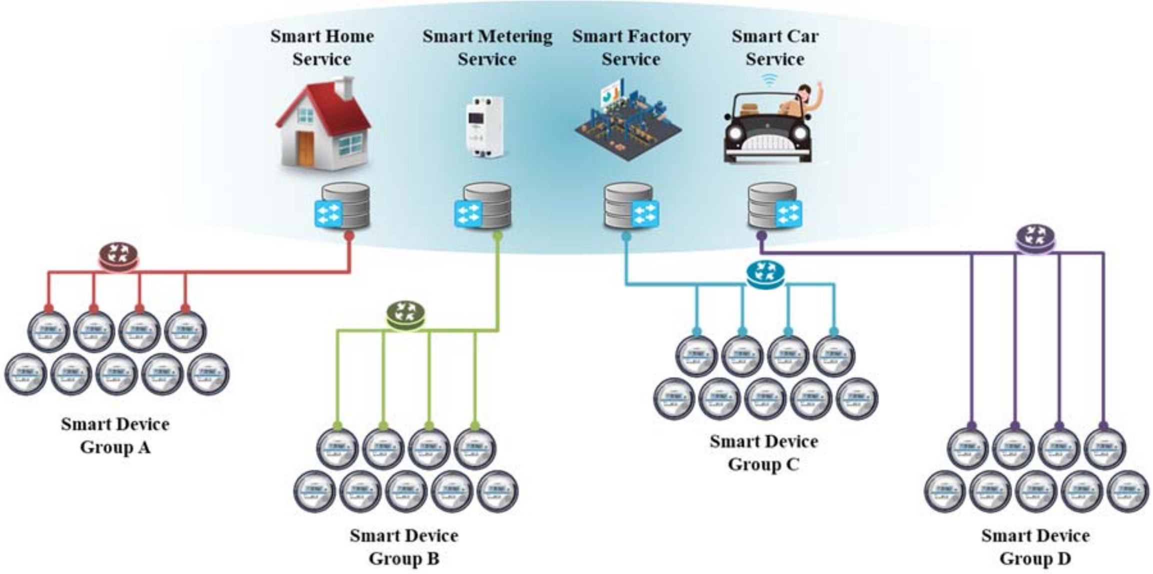

Fig.1. Group-based Service Network in IoT Environment

gateway of a smartphone, which can provide (for example) health information based on the data received. In such environments, devices that collect and process data, such as smartphones, connect with sensor devices to form a single small-device group, which in turn communicates with a service provider to provide customized services, or with other storage devices or groups of such devices. Likewise, a smart home may be a member of a group communicating with a service provider to update the group. The smart home may also communicate with neighbors, or with another group of users in a WBAN environment. However, if the number of devices in a group increases, or when several devices seek to communicate simultaneously either within or outside the group, each communication must be authenticated by the group leader employing group authentication technology [1], which is very efficient.

Here, we develop a group authentication and key exchange scheme that operates efficiently even within the lightweight devices of an IoT environment. First, as the number of devices in the group increases, the operation overhead imposed on the group leader becomes very large. The group leader performs authentications using a symmetrical key method (the keys must agree), generating a group key based on an XOR operation. We develop a group authentication/key agreement check method that greatly reduces the work overhead. We also develop a method that can reduce memory overhead, although the computation requirement is greater than that of our XOR-based method. We use a CRT-based secret-sharing technique to generate group keys.

2. RELATED RESEARCHES

2.1 Secret Sharing

The secret sharing technique has long been used to manage confidential information. A decryption key is required to access encrypted data. If the key is inadvertently lost or destroyed, the encrypted information can never be decrypted. Secret sharing was developed to prevent such scenarios.

Common secret sharing schemes employ the (t, n)-threshold form of secret sharing introduced by Shamir. Encrypted data can be restored only if at least t of the n participants sharing the secret information so request. In this scheme, secret sharing is achieved using polynomials on a finite field, and keys are restored between polynomials via Lagrangian interpolation. Other secret sharing techniques use either polynomials or various mathematical principles such as XOR[2], CRT(Chinese Remainder Theorem)[3,4], geometry, or one-way functions.

In 2009, Kurihara proposed a modified XOR-based secret sharing scheme that was a development of the (t, n)-threshold of Shamir [2]. Secret information n_p is saved in the form s $\left\{s_1, s_2, \ldots, s_{n_p-1}\right\} \in \{0,1\}^d$. A person seeking to access the secret information collects the distributed secret values generated via XOR from other persons in the network to restore s.

2.2 Group Authentication

In an IoT network, many users and devices are distributed in a non-centralized manner, which is associated with problems in authentication, access control, identification management, security, heterogeneous communications, and resource constraints. In particular, authentication and identification management is very important in the IoT context, and must be simple, safe, and fast [5,6].

Also, in a distributed environment, several devices may form a group, and secure within-group communication is also essential. Group authentication techniques explore whether a person or a device belonging to that person is actually a member of the group. The group leader then informs all devices in the group that the applicant is or is not a group member. Intra-group authentication (or group authentication; [7]) secures communications between groups using only a single authentication shared by the group leaders. Authentication is triggered when a device in either group receives or requests communication.

The group authentication scheme is classified into a method of having an authentication server and a method of without an authentication server. The way in which having an authentication server is present is typically the extensible authentication protocol (EAP) introduced in the IEEE 802.1x standard. EAP can be used for wireless ad hoc networks or mobile users.

In the case of a scheme without an authentication server, a group leader is generally determined in the group, and it is possible to perform the joining and leaving of devices in the group. Previously proposed group authentication schemes were based on the threshold secret sharing. Group authentication is not only performing authentication of participant in a group, but also provide of group-based authentication using a group key in inter-group authentication [1,5,6,7].

A group authentication scheme without an authentication server is typically a group authentication scheme (GAS) proposed by Harn for the first time. This scheme uses the Shamir-threshold secret sharing scheme. If m users with a threshold of

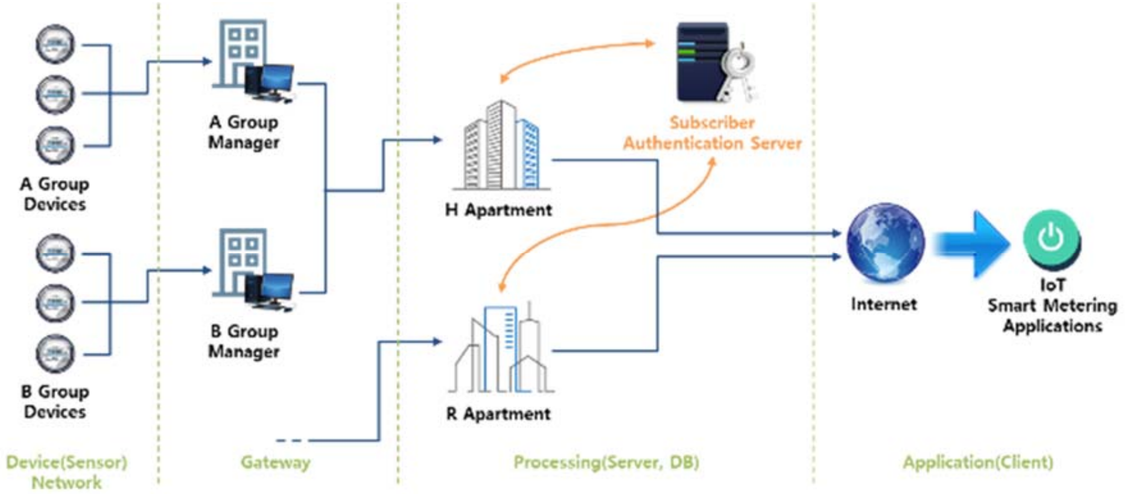

Fig.2. Group Authentication with an Authentication Server

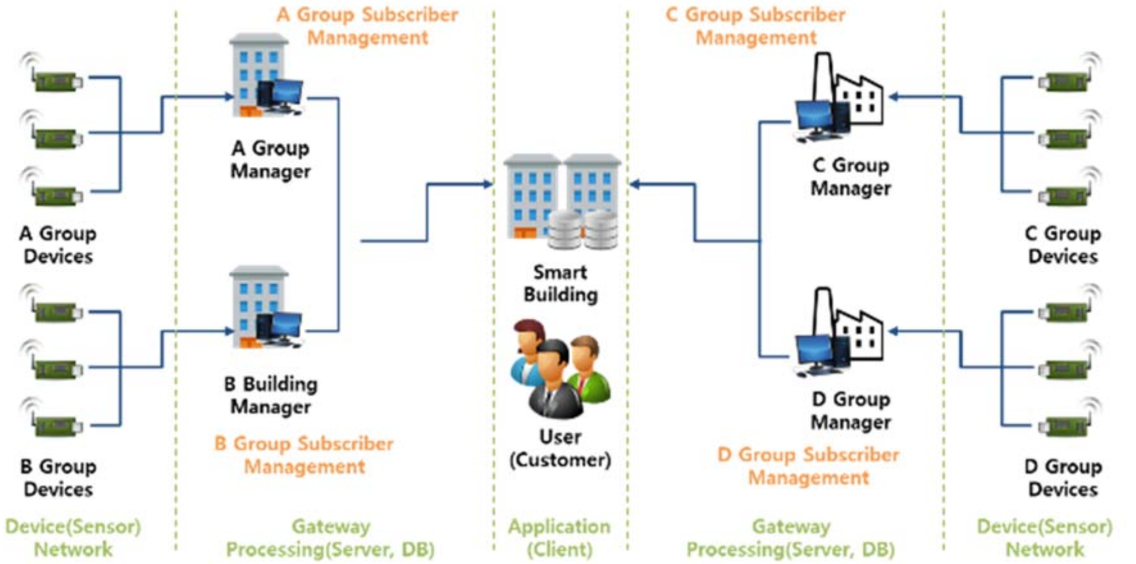

Fig.3. Group Authentication without an Authentication Server

t or more among n users participate in group authentication, the group authentication is successful. In this paper, we propose a new method for asynchronous (t, m, n) group authentication (GAS1) and asynchronous We propose a branch method. However, in GAS1, once used for authentication, the recovered secret is no longer a secret, so there is a problem that can only be used as a disposable one, raising a problem of reuse. GAS2 has been proposed as a group authentication technique with multi-usable characteristics to solve the above problem. However, in GAS2, it is possible to collect public tokens to check the secret value and use it to calculate the secret value of other participants to perform a compromise attack.

2.3 The Authenticated Key Exchange(AKE) Protocol

In 1992, Bellovin and Merritt developed a symmetrical encrypted key exchange (EKE) protocol using public keys [8] and proposed that the protocol could be used both for authentication and session key assignment. In the 2000s, Bellare and Rogaway developed the authenticated key exchange (AKE) protocol, which is not susceptible to dictionary attack [9]. The AKE protocol enables mutual authentication of a server and a client and creates session keys ensuring secure communication. Recently, a password-based AKE protocol (PAKE) has also been developed, using passwords with relatively low entropies.

Here, we use an intra-group AKE protocol to perform both intra- and inter-group authentication in conjunction with a service provider.

3. SECURITY REQUIREMENTS

In this chapter, we analyze the IoT security requirements that group AKE schemes must meet. There must be no limit on the number of group members, but they must be appropriately managed. Also, the scheme overload should be low, to prevent retransmission attacks and to adapt to changes in the IoT [1].

· **Authentication:** The most important security requirement of the IoT is authentication. Existing methods feature mutual authentication or three-way authentication involving a server. As the IoT environment gradually changes from centralized to grouped, both inter-object and inter-group authentication are required. The group leader must confirm that would-be communicators are indeed both group members and legitimate users.

· **Prevent Replay Attacks:** In the IoT, both people and devices access the Internet wirelessly. When a user or device requests authentication, an attacker may seek to intercept and retransmit the message to the group leader, thus accessing the group and perhaps sending malicious messages to group members.

· **Efficiency:** As the IoT fosters communications between groups, the larger the group, the more computation time is required by the group leader to authenticate participants, associated with increasing overhead. Therefore, group authentication and key exchange must maximize the efficiency of the group leader in this context.

· **Identify Malicious Participants:** Malicious participants inside the group should not be able to perform the authentication process. To do this, as in the case of solving the problem in the existing Harn method, the information created based on the identifier must be verified during the authentication process. Although the Chien [11] method solves the problem, it is difficult to find malicious group leaders even though they can find malicious participants because of their different communication structures.

4. PROPOSED SCHEMES

In this chapter, we analyze the IoT security requirements that group AKE schemes must meet. There must be no limit on the number of group members, but they must be appropriately managed. Also, the scheme overload should be low, to prevent replay attacks and to adapt to changes in the IoT [1].

4.1 System Parameters

The parameters used are as follows:

ID_* : Identifier of $*$
$h(\cdot)$: One way hash function
s : Group secret key
k : Threshold for restoring the group secret key
w : Distributed secret array
R : Distributed secret modulus

4.2 XOR-based Group Authentication

Here, we develop an efficient XOR-based group AKE scheme for the IoT environment. The scheme features intra-group authentication and can be extended to include group authentication by a service provider.

4.2.1 Joining phase

Step 1. The group leader generate a secret key x.

Step 2. Node i (a group member) transmits its identifier ID_i to the leader.

Step 3. The leader computes a secret key $b_i = h(x \parallel ID_i)$ based on the node identifier to confirm that node i is a member of the group and ensures that b_i is included in the bloom filter B. If b_i is already in bloom filter B, the request is ignored.

Step 4. The leader generates a group secret key $s = \left\{s_0 \parallel s_1 \parallel \cdots \parallel s_{n_p-1}\right\}$ if node i is a new node.

$$\left(s \in \{0,1\}^{d(n_p-1)},\ s_0 = 0^d\right)$$

Step 5. The leader generates a two-dimensional random number array $r_j^i = rand(\{0,1\}^d)$ that, in turn, creates a distributed secret value for each node.

$$(0 \leq i \leq k-2,\ \ 0 \leq j \leq n_p - 1)$$

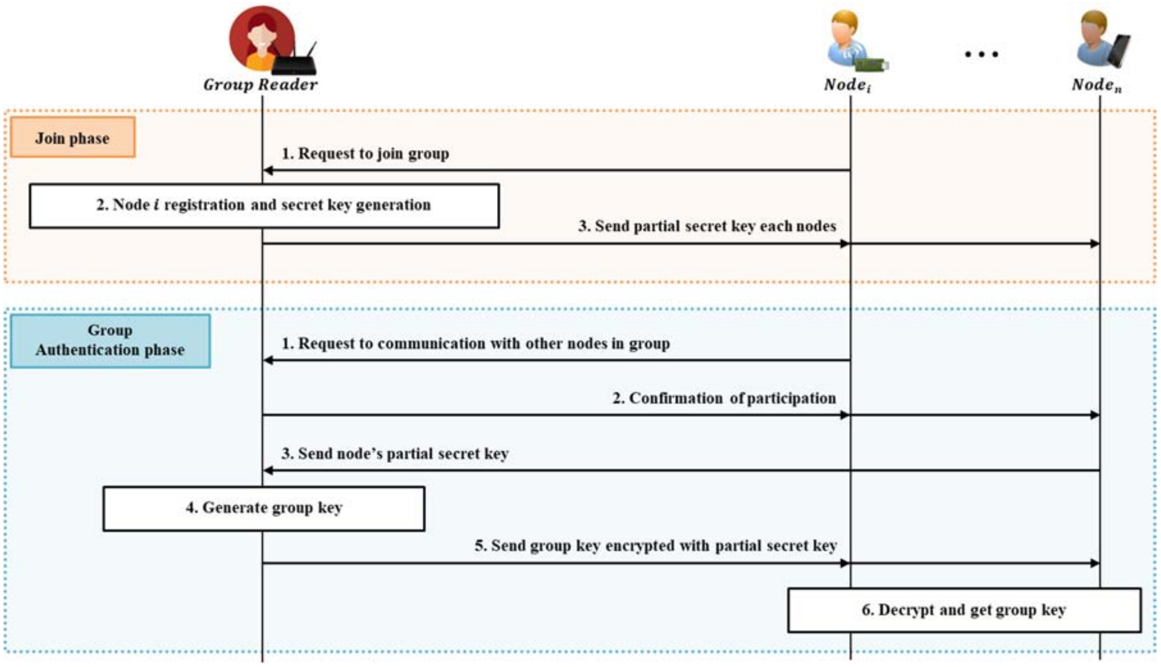
Group Reader
$Node_i$
$Node_n$
Join phase
1. Request to join group
2. Node i registration and secret key generation
3. Send partial secret key each nodes
Group Authentication phase
1. Request to communication with other nodes in group
2. Confirmation of participation
3. Send node's partial secret key
4. Generate group key
5. Send group key encrypted with partial secret key
6. Decrypt and get group key

Fig.4. Scenario of Proposed Group Authentication and Key Exchange Schemes

Step 6. The leader computes $w_{(i,j)} = \left(\bigoplus_{h=0}^{k-2} r_{h\cdot i+j}^{h}\right) \oplus s_{j-i}$ using a random number array r_j^i.

Step 7. The leader computes the distributed secret value $w_i = \left\{w_{(i,0)} \parallel \cdots \parallel w_{(i,n_p-2)}\right\}$ for each node i. If w_i is the threshold value k, the group leader can recover the group secret value s.

Step 8. The leader sends $(w_i \parallel b_i)$ to each node of the group, and the nodes store these values.

4.2.2 Authentication request phase

Step 1. Node i generates an arbitrary number r (a request for generation of a session group key).

Step 2. Node i generates $E_{b_i}(w_i \parallel ID_i \parallel r)$ using b_i, which is the secret key for this node stored in the joining step, and transmits this value to the leader.

Step 3. The leader receives the authentication request of node i, and decodes the received value to generate $b_i = h(x \parallel ID_i)$. After confirming that b_i is included in the counting bloom filter B, the leader sends a join "wait signal", and r, to the nodes that will participate in group authentication.

Step 4. Each node sends its $E_{b_n}(w_n \parallel ID_n \parallel r), ID_n$ encrypted with its secret key to the leader. At this time, to restore the secret value s, at least k nodes must participate.

4.2.3 Group authentication and key exchange phase

Step 1. The leader computes the secret key $b_i' = h(x \parallel ID_i)$ of all nodes that are in communication and confirms $b_i' \in B$ for each b_i'. Then, if each node is in fact a member of the group, the encrypted value is decrypted and its validity is verified by checking the r value.

Step 2. The leader calculates the $w_{t_i} \rightarrow \left\{w_{(t_i,0)} \parallel \cdots \parallel w_{(t_i,n_p-2)}\right\}$ array using the $\left\{w_{t_0}, w_{t_1}, \ldots, w_{t_{k-1}}\right\}$ set received from k nodes.

Step 3. The leader calculates an array w allowing recovery of the secret value, as follows:

$$w = \left(w_{(t_0,0)}, \ldots, w_{(t_0,n_p-2)}, \ldots, w_{(t_{k-1},0)}, \ldots, w_{(t_{k-1},n_p-2)}\right)^T$$

Step 4. The leader computes the matrix M as follows, and then $M \cdot w$, to recover the secret value

s . The MAT function is that proposed by Kurihara[2].

$$M = MAT(t_0, \dots, t_{k-1})$$
$$M \cdot w \rightarrow \left(s_1, \dots, s_{n_p-1}\right)^T$$
$$s = \left(s_1 \parallel \cdots \parallel s_{n_p-1}\right)$$

Step 5. The leader generates a group session key $SK = h(s \parallel r)$.

Step 6. The leader sends $E_{b_i}(SK \parallel r)$ to n nodes, thus distributing the generated session key, decrypts communications received from nodes using the node's secret key, and verifies r. This generates the session key SK.

4.3 CRT-based Group Authentication

Here, we develop a CRT-based group AKE scheme for the IoT environment. As with the XOR-based group AKE developed above, an intra-group authentication method is designed first, but this can be extended to higher-level group authentication by the service provider.

4.3.1 Joining phase

Step 1. The group leader generates a secret key x.

Step 2. Node i (a group member) transmits its identifier ID_i to the leader.

Step 3. The leader computes a secret key $b_i = h(x \parallel ID_i)$ based on the node identifier to confirm that node i is a member of the group, and ensures that b_i is included in the bloom filter B. If b_i is already in bloom filter B, the request is ignored.

Step 4. The leader generates distributed secret modulus $(R_1 \parallel R_2 \parallel \cdots \parallel R_n)$ for each participating node, if node i is a new node.

$$(\gcd(R_i, R_j) = 1, i \neq j)$$

Step 5. The leader uses the CRT with distributed secret modulus to calculate the group secret key s as follows.

$$b_1 \; mod \; R_1 \equiv s$$
$$b_2 \; mod \; R_2 \equiv s$$
$$b_3 \; mod \; R_3 \equiv s$$
$$\vdots$$
$$b_n \; mod \; R_n \equiv s$$

Step 6. The leader computes the distributed secret modulus R_i for each node i. If R_i is the threshold value k, the group leader can recover the group secret value s.

Step 7. The leader sends $(R_i \parallel b_i)$ to each node of the group, and the nodes store these values.

4.3.2 Authentication request phase

Step 1. Node i generates an arbitrary number r (a request for generation of a session group key).

Step 2. Node i generates $E_{b_i}(R_i \parallel ID_i \parallel r)$ using b_i, which is the secret key for this node stored in the joining phase, and transmits this value with ID_i to the leader.

Step 3. The leader receives the authentication request of node i, and decodes the received value to generate $b_i = h(x \parallel ID_i)$. After confirming that b_i is included in the counting bloom filter B, the leader sends a join "wait signal", and r, to the nodes that will participate in group authentication.

Step 4. Each node sends its $E_{b_n}(R_n \parallel ID_n \parallel r), ID_n$ encrypted with its secret key to the leader. At this time, to restore the secret value s, at least k nodes must participate.

4.3.3 Group authentication and key exchange phase

Step 1. The leader computes the secret key $b_i' = h(x \parallel ID_i)$ of all nodes that are in communication and confirms $b_i' \in B$ for each b_i'. Then, if each node is in fact a member of the group, the encrypted value is decrypted, and its validity is verified by checking the r value.

Step 2. The leader calculates the s' as follows using the $\{R_1, R_2, \dots, R_k\}$ received from k nodes.

$$s' \equiv b_1 \; mod \; R_1$$
$$s' \equiv b_2 \; mod \; R_2$$
$$s' \equiv b_3 \; mod \; R_3$$
$$\vdots$$
$$s' \equiv b_k \; mod \; R_k$$

Step 3. If s' is generated correctly by comparing the s stored in the group leader, the participating nodes are authenticated.

Step 4. The leader generates a group session key $SK = h(s \parallel r)$.

Step 5. The leader sends $E_{b_i}(SK \parallel r)$ to k nodes, thus distributing the generated session key, decrypts communications received from nodes using the node's secret key, and verifies r. This generates the session key SK.

5. ANALYSIS OF PROPOSED SCHEMES

Here, we develop a group AKE scheme that operates efficiently in the lightweight devices of IoT environments. As the number of devices in a group increases, the operation overhead assumed by the group leader becomes very large. Therefore, the group leader engages in symmetrical key authentication (key agreements) and generates a group key based on an XOR operation. Group authentication and key agreements reduce the work overhead. Second, we develop a method that reduces memory overhead, although the computation requirement is higher than the XOR-based method. We generate group keys using a CRT-based secret sharing technique. This Chapter confirms that both approaches meet the security requirements mentioned in Chapter 3.

5.1 XOR-based Group Authentication

The efficient XOR-based group AKE scheme for the IoT environment proposed in Section 4 is an intra-group authentication method that can be extended by the service provider to allow group authentication.

· **Authentication:** A group member sends information to the leader, who uses the identifier ID_i and node's key b_i to determine whether the member is in the bloom filter, and, if so, authenticates the member. Thus, the member has now passed the joining step.

· **Prevent Replay Attack:** As an attacker does not own the secret key b_i generated in the authentication phase, s/he cannot attack using a self-generated r. Also, even if the attacker intercepts and retransmits the intermediate values created in the authentication request and the group AKE phases, the attacker cannot obtain the session key SK because s/he does not own the distributed secret value w_i.

· **Efficiency:** The method uses an XOR operation to calculate the group secret value s employing information on group participants created during group AKE, and compares s with the value obtained using a secret restoration process employing an

Table.1. Analysis of Proposed Schemes

	[5]	[6]	[7]	[8]	Proposed Scheme 1	Proposed Scheme 2
Base System	Paillier Threshold	Nyberg Accumulator	Shamir Threshold	IBS	**XOR**	CRT
Environments	IoT	IoT	Group Communication (D2D)	Ad Hoc Network	IoT	IoT
Group Limit	Unlimited	Unlimited	Limit	Unlimited	Unlimited	Unlimited
Dynamic Member Management	Can dynamically add and remove members	Can dynamically add and remove members	Can dynamically add and remove members	Weak	Can dynamically add and remove members	Can dynamically add and remove members
Replay Attacks	Use new values per session	Weak	Weak	Use signature	Use new values per session	Use new values per session
Minimum Number of Participants	Threshold k	All participants	Threshold k	Dynamic participation	Threshold k	All participants
Group Leader Memory Overhead	High Device per 128bytes	Low(Hash based) Fixed size 128bytes	High Device per 128bytes	High(ID based) Device per 256kb	High(XOR based) Device per 256bytes	Low(Hash based) Fixed size 128bytes
Device Operation Overhead	High(Signature based) NPE	Low(Hash based) N(SE+⊕)	Low(Threshold based) NSE	High(Bilinear DH based) N(eP+bP)	Low(XOR based) NSE	Low(CRT based) N(SE+⊕)
Complexity	High log($klog^2k$)	Low log(n)	High log($klog^2k$)	High log(n^2)	Low $log(n)$	Low $log(n)$

SE : Symmetric Encryption(Decryption), PE : Public Key Encryption
eP : Elliptic Curve Addition Operation, bP : Bilinear Pairing Operation
⊕ : XOR Operation , N : Number of Nodes

existing public key distribution scheme. Efficiency is thus assured.

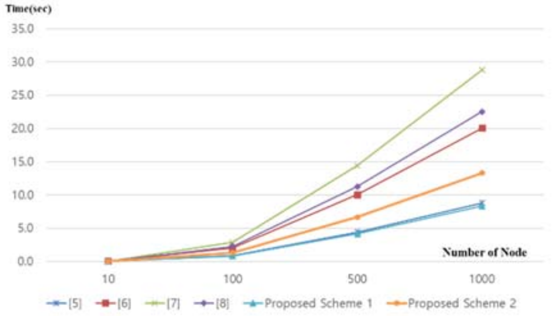

Fig.5. Comparison of Group Authentication and Key Exchange Computation by Number of Nodes

5.2 CRT-based Group Authentication

Section 5 proposes a CRT based group authentication and key exchange scheme for the proposed IoT environment. Like the XOR-based group authentication and key exchange, the Intra-group Authentication method is designed first, and it can be extended to the higher group authentication through the service provider.

· **Authentication:** As with the XOR-based group authentication scheme, the group leader checks that node's identifier ID_i and key b_i to determine whether the member is in the bloom filter. Thus, the member has now passed the joining step.

· **Prevent Replay Attacks:** When a node requests authentication, it selects an arbitrary r, and then includes r in the authentication process or the generation of the session key. If the attacker intercepts the authentication request of the node and retransmits it later, it can not perform the authentication process because it does not know the secret key b_i, and can not generate the session key.

· **Efficiency:** When a group leader creates a distributed secret modulus and creates a group secret key, it uses the rest of the CRT. This can reduce the amount of computation for key generation while taking up a relatively small amount of memory. However, when generating the distributed secret modulus, each value must be coprime that the key generation can take a long time, but the verification process is quick.

6. CONCLUSIONS

We developed a symmetric key authentication scheme (a key agreement) implemented by a group leader, and a group key generation method based on an XOR operation, to ensure group AKA. As the number of devices in the group increases, the operation overhead of the group leader becomes very large. Our method is associated with a lower computational overhead than existing threshold-based group authentication methods. In addition, as the XOR-based group authentication scheme carries a high memory overhead, we also develop a CRT-based secret sharing scheme to reduce this overhead.

Group authentication schemes are being vigorously researched because of recent developments in IoT environments. In future, it will be necessary to reduce the memory overhead of the group leader and the computational loads of devices.

ACKNOWLEDGMENTS: This research was supported by the MSIT(Ministry of Science, ICT), Korea, under the ITRC(Information Technology Research Center) support program (IITP-2017-2015-0-00403) supervised by the IITP (Institute for Information & communications Technology Promotion)

REFRENCES:

[1] W. T. Su, W. M. Wong, and W. C. Chen, "A survey of performance improvement by group-based authentication in IoT," *Applied System Innovation (ICASI) 2016 International Conference on. IEEE*, pp. 1-4, (2016)

[2] J. Kurihara, S. Kiyomoto, K. Fukushima, and T. Tanaka, "A new (k, n)-threshold secret sharing scheme and its extension," *Information Security*, pp. 455-470, (2008)

[3] R. Steinfeld, P. Josef P, and W. Huaxiong, "Lattice-based threshold-changeability for standard CRT secret-sharing schemes," *Finite Fields and Their Applications*, Vol. 12, No. 4, pp. 653-680, (2006)

[4] YV. S. Rao, and C. Bhagvati, "CRT based threshold multi secret sharing scheme," *International Journal of Network Security*, Vol. 16, No. 3, pp. 194-200, (2014)

[5] P. N. Mahalle, N. R. Prasad, and R. Prasad, "Threshold cryptography-based group authentication (TCGA) scheme for the internet of things (IoT)," Wireless Communications, *Vehicular Technology, Information Theory and Aerospace & Electronic Systems, 2014 4th International Conference on. IEEE*, pp. 1-5, (2014)

[6] J. J. Huang, W. S. Juang, C. I. Fan, Y. F. Tseng, and H. Kikuchi, "Lightweight authentication scheme with dynamic group members in IoT environments," *Adjunct Proceedings of the 13th International Conference on Mobile and Ubiquitous Systems: Computing Networking and Services*, pp. 88-93, (2016)

[7] L. Harn, "Group authentication," IEEE Transactions on computers, Vol. 62, No. 9, pp. 1893-1898, (2013)

[8] F. Wang, C. C. Chang, and Y. C. Chou, "Group Authentication and Group Key Distribution for Ad Hoc Networks," *International Journal of Network Security*, Vol. 17, No. 2, pp. 199-207, (2015)

[9] S. Bellovin, and M. Merritt, "Encrypted key exchange: password-based protocols secure against dictionary attacks," *IEEE Symposium on Research in Security and Privacy*, pp. 72-84, (1992)

[10] M. Bellare, D. Pointcheval, and P. Rogaway, "Authenticated key exchange secure against dictionary attacks," *EUROCRYPT 2000, Lecture Notes in Computer Science*, Vol. 1807, pp. 139-155, (2000)

[11] H. Y. Chien, "Group Authentication with Multiple Trials and Multiple Authentications," *Security and Communication Networks* 2017, 2017.

[12] D. H. Lee and I. Y. Lee, "Efficient Group Authentication and Key Exchange Scheme for IoT Environment", *Conference of the IWCIT*, 2017.