symmetry

Article

An Efficient Secure Scheme Based on Hierarchical Topology in the Smart Home Environment

Mansik Kim [1], Kyung-Soo Lim [2], Jungsuk Song [3] and Moon-seog Jun [1,*]

[1] Department of Computer Science and Engineering, Soongsil University, Seoul 07027, Korea; mansik@ssu.ac.kr
[2] Intelligent Security Research Group, Electronics and Telecommunications Research Institute, Daejeon 34129, Korea; lukelim@etri.re.kr
[3] Department of Advanced KREONET Security Service, Korea Institute of Science and Technology Information, Daejeon 02792, Korea; song@kisti.re.kr
* Correspondence: mjun@ssu.ac.kr; Tel.: +82-2-826-6526

Received: 5 July 2017; Accepted: 31 July 2017; Published: 3 August 2017

Abstract: As the Internet of Things (IoT) has developed, the emerging sensor network (ESN) that integrates emerging technologies, such as autonomous driving, cyber-physical systems, mobile nodes, and existing sensor networks has been in the limelight. Smart homes have been researched and developed by various companies and organizations. Emerging sensor networks have some issues of providing secure service according to a new environment, such as a smart home, and the problems of low power and low-computing capacity for the sensor that previous sensor networks were equipped with. This study classifies various sensors used in smart homes into three classes and contains the hierarchical topology for efficient communication. In addition, a scheme for establishing secure communication among sensors based on physical unclonable functions (PUFs) that cannot be physically cloned is suggested in regard to the sensor's low performance. In addition, we analyzed this scheme by conducting security and performance evaluations proving to constitute secure channels while consuming fewer resources. We believe that our scheme can provide secure communication by using fewer resources in a smart home environment in the future.

Keywords: emerging sensor network (ESN); hierarchical topology; security; smart home; physical unclonable functions (PUFs); Internet of Things (IoT)

1. Introduction

Due to the development of the IoT technology, people can receive service via the Internet at any time and from any place [1,2]. IoT has been used in various fields, including theoretical technologies. For example, it has been applied in the smart home environment, which provides many services [3–5]. According to Strategy Analytics, the global smart home market has been growing by 19% on an annual average. It is expected for the scale of the market to reach $115 billion in 2019 [6]. In addition, according to Harbor Research, the number of IoT devices to be installed around the world is expected to be 8 billion, and 47% of them are expected to be installed in smart homes. Smart home service is an ESN in which integrates IoT emerging technologies such as autonomous driving, cyber physical system and mobile nodes, and existing sensor networks has been in the limelight [7]. Therefore, many of the companies and research institutions have been developing diverse technologies so as to provide better services [8,9]. However, a smart home combined with a sensor network still faces problems in terms of low power and security. Various studies have been conducted to solve these issues [10,11]. In February 2015, HP indicated in their research reports that most of the smart home IoT devices were weak in encrypting passwords and in the authentication procedures in their research reports. They warned that there was a high possibility of users being exposed to cyber-crime as personal information is required to use smart home IoT devices. In addition, according to Symantec's report on the status of smart home device security, a weakness in authentication exists [12,13]. Especially, sensor network topology needs to be taken into account in order to efficiently and securely deliver information while minimizing the electronic consumption of sensors in ESNs. However, various security techniques and topologies for previous sensor networks have not addressed the diverse capabilities of smart home sensors. Therefore, they are inefficient or inappropriate for a low-power sensor network [14–16]. In this study, we propose a security technique in hierarchical topology for smart home sensor networks that has various capabilities. This study is laid out as follows: Section 2 describes the infrastructure of a smart home, the security requirements for a smart home, and previous research on smart homes. Section 3 describes the mutual authentication proposed in this study and techniques for establishing a security channel in detail. Section 4 provides a security evaluation of the suggested scheme, computing resource analysis, and storage resource analysis. Finally, Section 5 provides our conclusions.

2. Related Works

In this section, we discuss the smart home infrastructure, features of smart home, and previous related works on these areas.

2.1. Infrastructure of a Smart Home

In general, the smart home infrastructure is comprised of a sensor network that has been created with various wireless sensors inside of the home, an AP for connecting the sensor network outside, and a service provider. Figure 1 shows the infrastructure of a typical smart home. Various sensors in the smart home communicate with each other through a sensor network to provide service to, and collect information from, residents. Each of the sensors exchanges information with the service provider through the AP and if a sensor cannot directly reach the AP, it communicates through the other sensors. A smart home sensor network requires a topology for securely and efficiently exchanging information in regards to computing ability and the power capacity of various sensors. However, most of the suggested sensor networks propose various topologies or infrastructures that lack a diversity in sensor ability. Therefore, it is inappropriate to apply them to an actual smart home sensor network.

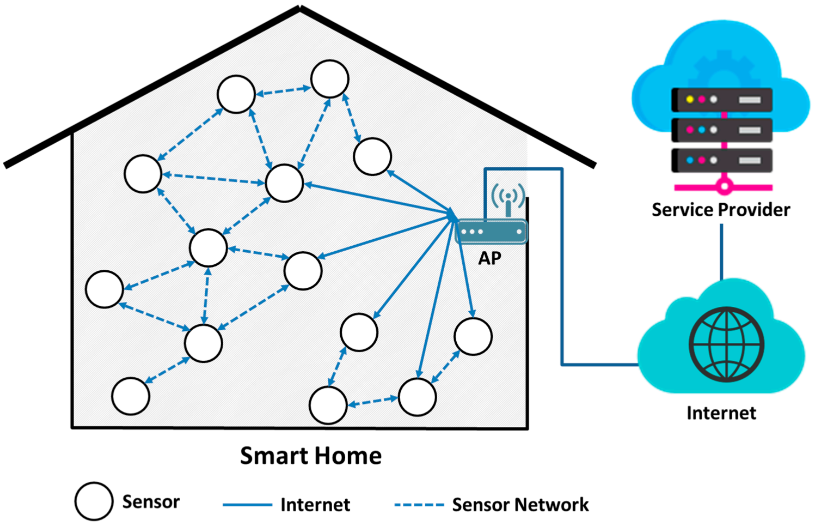

Figure 1. Existing topology in a smart home sensor network.

2.2. Features of Smart Home

In the smart home environment, it is required to satisfy security demands and have the appropriate topology model to securely and efficiently exchange information among sensors, or between the sensor and service provider. In addition, it is necessary to appropriately utilize various sensor capabilities when using security techniques and to, especially, consider sensors with low resources.

2.2.1. Topology for ESNs in smart homes

Most of the sensors with low computing ability and power are arranged and distributed in wireless sensor networks (WSNs). Therefore, it is required to establish a plan for securely and efficiently delivering information. In order to solve problems in the sensor network, various topologies have been suggested and applied in current services [17–19]. However, most of the topologies suggested in WSNs did not take particular environments or the various abilities of sensors into consideration. A smart home sensor network is made up of ESNs where the IoT environment and previous sensor network are combined, unlike in existing WSNs. Sensors in a smart home are structured to communicate with an external network through the AP and the available distance of communication and computing abilities are different in each sensor. The topology suggested in the existing WSNs has not taken the smart home environment into account and, hence, is inefficient. In order to efficiently exchange information in the smart home sensor network, a topology that takes the capabilities of various sensors into account is required.

2.2.2. Security

Due to the development of IoT technology, the number of sensors has been exponentially increasing and used in various fields. A smart home is a representative sensor network that is combined with IoT technology in order to distribute diverse sensors in the home so as to provide convenience to the inhabitants. Most of the information collected from these sensors is particular to the users. If they are attacked by malicious attackers, there is a possibility for them to have their privacy significantly invaded, their lives to be threatened, and the loss of property [20]. In order to provide users a service that is secure against these threats, smart home sensors shall carry out mutual authentication prior to exchanging information with an external network and establish secure communication by exchanging keys. In addition, they need to be against various malicious attacks, including relay attacks, replay attacks, leaked keys, and forward secrecy [21]. In order to establish a secure sensor environment, efficient large-group key (ELK) distribution has been suggested by Tien-Dung et al. [22] and three strategies for securely distributing rekey messages has been proposed by Mohammad et al. [23]. However, none of these are appropriate in the smart home environment. Adrian et al. [24] proposed the combining group-key and time-key (CoGKTK) for securing multi-cast techniques in a sensor environment. Wong et al. [25] proposed the usage of statistical group index matching (SGIM), which is not secure against various security threats.

2.2.3. Different sensor performances

A smart home sensor network is comprised of various sensors. For example, sensors installed in large products, such as refrigerators or washing machines, can possess a relatively higher computing ability and power than smaller ones, such as thermometers and pots. Since the capabilities possessed by sensors depend on what their purpose is for a product, the aforementioned environment must be considered for information to be efficiently exchanged in the smart home sensor network.

2.2.4. Low resource

In order for each sensor to securely exchange information in the smart home sensor network, it is required to establish mutual authentication and a security channel among sensors. However, most of the sensors only took size or price into account and used low-capacity chips or small batteries [26–28]. Sensors installed in small products possess relatively low resources. Sensors with low resources are not able to perform complicated calculations for secure communication and have fewer values to preserve due to having a low storage capacity. Therefore, a lightweight scheme is required so that sensors with low resources can securely communicate.

2.3. Previous Studies on Smart Homes

In this section, previous studies related to smart home service and security are reviewed. Alessandro et al. [29] proposed a wireless architecture that estimates the presence, movements, and behaviors of elders who reside in a smart home by monitoring and managing its power. The suggested wireless architecture is a flexible wireless architecture that satisfies user acceptance and system performance through the amount of large-scale data collection and training based on the locations and behaviors of those who reside in the home. In addition, developers can access physical data without having to worry about hardware capabilities by abstracting information that has been collected from different devices through an abstraction layer of a software stack in a multi-platform environment with heterogeneous wireless devices. Therefore, service is provided from the upper layers. In addition, integrated and low-cost wireless architecture has been suggested for guaranteeing two important key points for smart homes in the future: user acceptance and low system complexity.

Vijay et al. [30] have provided network-level protections to monitor network activity and detect suspicious behaviors as a solution for privacy or security problems that snoop, or intrude, on a family's activities. This is necessary because smart home appliances, such as smoke alarms, power switches, and baby monitors, have increased exponentially. In the suggested scheme, software defined networking (SDN) that can dynamically block/quarantine devices based on the home, such as time of day, occupancy, or network activity, and it serves as a dynamic security rule. In addition, they proposed an external entity for the security management provider (SMP) that develops, customizes, and delivers extra safeguards in the network level for users' smart home devices. Therefore, they suggested a three-party architecture comprised of the SMP role, ISP/home-router-vendor role, and customer role that provide security as a service.

Debraj et al. [31] proposed a system that can monitor and survey residents according to information collected from WNS in the smart home environment. They installed various sensors at home where actual residents live to monitor their behaviors through smart home sensors and to collect information for six months. The topology and connected information of each sensor was collected in the central station, and this information has been used for households and industry applications. Since this test was performed on an actual residential environment, it was feasible to collect information and analyze them in the setting that was similar to an actual service environment. It was also possible to experiment with and monitor various sensors used in the smart home.

Basma et al. [32] proposed a smart home wireless biometric smart home (WB-SH) design in the use of a wireless sensor network and biometric technologies. WB-SH uses a wireless sensor and actuator network (WSAN) that senses and performs work while operating the smart home. They also used bio-information and reinforced smart home security. In addition, they used sensors with a large amount of power in order to perform the heavy work in accordance with the location of each sensor or power supply source.

3. Proposed Infrastructure

3.1. Proposed Sensor Network Topology of a Smart Home

Figure 2 shows the hierarchical topology that is suggested in this paper in consideration of the capabilities of various sensors in a smart home environment. Sensor networks in the smart home have been classified into three classes of low-, middle-, and high-class from sensors with low resources to ones with high resources for the capabilities of each sensor. Low-class sensors are the smallest and most affordable. Therefore, they are distributed the most in the smart home. Middle-class sensors are not distributed as widely as low-class sensors. In addition, high-class sensors have the highest capabilities, and one of them is placed in each smart home. These sensors do not communicate with the one that is nearest, but with the closest high-class sensors, in order to exchange information with service providers. For example, low-class sensors only communicate with the closest middle-class sensor, and middle-class sensors only communicate with the nearest low-class and high-class sensors.

Direct communication with the service provider through an AP is only carried out by high-class sensors. Suggested techniques in this paper perform mutual authentication and key agreements, as all the sensors have PUFs, and they implement them for secure communication amongst the layers in each sensor. PUFs are a unique chip that cannot be physically copied. As such, they present a unique challenge-response value. The challenge-response value of each PUF is registered in the server's PUF DB before sensors are distributed to the field [33–36]. Values that are used once are removed from the PUF's DB. Therefore, they cannot be re-used.

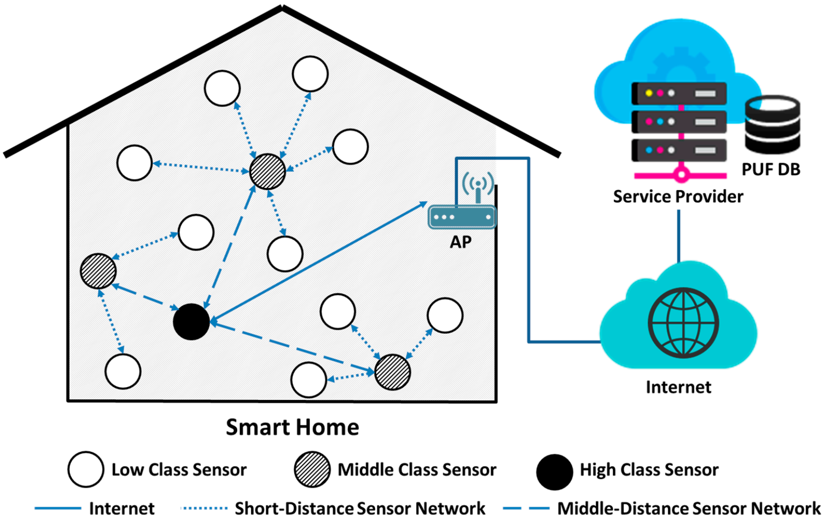

Figure 2. Proposed hierarchical topology in a smart home sensor network.

3.2. Proposed Protocols

A key agreement technique and a mutual authentication for our proposed lightweight hierarchical topology are divided into a provisioning phase, mutual authentication phase, and key agreement phase. In the provisioning phase, sensors in all the classes are registered in the service provider before they are distributed in a smart home. In the mutual authentication and key agreement phases, sensors in each class perform mutual authentication and a key agreement with the service providers and high-class sensors. Low-class sensors are required to go through high-class sensors to communicate with service providers. Therefore, mutual authentication and the key agreement are first performed by high-class sensors. The parameter for the proposed protocol is described in Table 1.

Table 1. Proposed protocol parameters.

Notation	Meaning
SP	Service Provider
Sensor	Sensors including HS, MS, and LS
HS	High-class sensor
MS	Middle-class sensor
LS	Low-class sensor
PUFs()	Physical unclonable functions
PUF DB	Challenge and Response value mapping DB for PUF()
E()	Encrytion function
h()	Hash function
ID	Identification
N	Randomly gerated Nonce
C	Challenge value for PUF
R	Response value for PUF
M	Encrypted Message
V	Verification Message
SK	Session Key
i	The number of sensor
j	The number of MS
k	The number of LS

3.2.1. Provisioning Phase

All of the sensors are registered in the service provider in the provisioning phase, as shown in Figure 3, before they are deployed in the smart home. In the provisioning phase, it is assumed that communication between the sensor and service provider has already been secured in the provisioning phase.

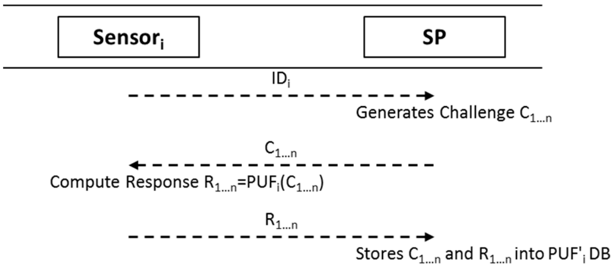
Sensor$_i$
SP
ID$_i$
Generates Challenge C$_{1…n}$
C$_{1…n}$
Compute Response R$_{1…n}$=PUF$_i$(C$_{1…n}$)
R$_{1…n}$
Stores C$_{1…n}$ and R$_{1…n}$ into PUF'$_i$ DB

Figure 3. Sensor provisioning phase.

Step 1. Sensor$_i$ sends their ID_i to SP.
Step 2. SP confirms ID received from Sensor$_i$ and generates n challenge values to create PUF_i DB in the Sensor$_i$ and sends them to Sensor$_i$.
Step 3. Sensor$_i$ received Challenge $C_{1\ldots n}$ from SP and computes Response $R_{1\ldots n}$ in correspondence with $C_{1\ldots n}$ by using the PUF_i chip that it owns and sends them to SP.
Step 4. SP received with Response $R_{1\ldots n}$ from Sensor$_i$ maps $C_{1\ldots n}$ and Response $R_{1\ldots n}$ in 1:1 and stores them in the PUF_i DB for a challenge-response with Sensor$_i$ in the future.

3.2.2. Authentication and Key Agreement Phases

Once the sensors are distributed in a smart home, high-class sensors first perform mutual authentication and a key agreement with service providers. Figure 4 shows the phase of mutual authentication and key agreement between high-class sensors and service providers.

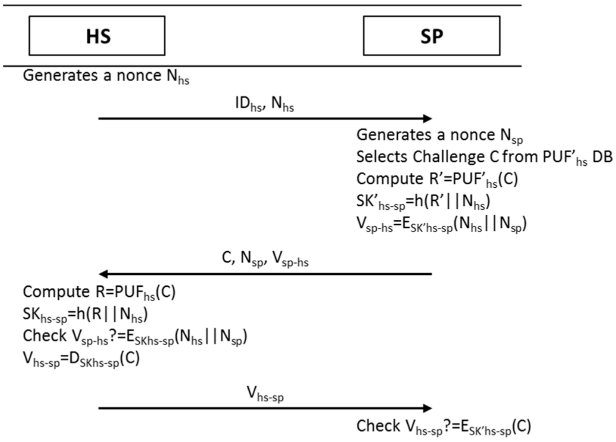
HS
SP
Generates a nonce N$_{hs}$
ID$_{hs}$, N$_{hs}$
Generates a nonce N$_{sp}$
Selects Challenge C from PUF'$_{hs}$ DB
Compute R'=PUF'$_{hs}$(C)
SK'$_{hs-sp}$=h(R'||N$_{hs}$)
V$_{sp-hs}$=E$_{SK'hs-sp}$(N$_{hs}$||N$_{sp}$)
C, N$_{sp}$, V$_{sp-hs}$
Compute R=PUF$_{hs}$(C)
SK$_{hs-sp}$=h(R||N$_{hs}$)
Check V$_{sp-hs}$?=E$_{SKhs-sp}$(N$_{hs}$||N$_{sp}$)
V$_{hs-sp}$=D$_{SKhs-sp}$(C)
V$_{hs-sp}$
Check V$_{hs-sp}$?=E$_{SK'hs-sp}$(C)

Figure 4. Authentication and key agreement phase for a high-class sensor.

Step 1. HS generates a random number N_{hs} and sends it to SP with its identifier ID_{hs}.
Step 2. SP received with N_{hs} and ID_{hs} from HS generates the random number N_{sp} and selects Challenge C from the PUF'_{hs} DB that is relevant to ID_{hs}. In addition, it computes the Response R' value corresponding to Challenge C from $PUF'_{hs}(C)$ and hashes $(R'||N_{hs})$ to create session key $SK'_{hs-sp} = h(R'||N_{hs})$. Then, SP connects the N_{hs} received from HS and N_{sp} to compute V_{sp-hs} $= E_{SK'hs-sp}(N_{hs}||N_{sp})$ the value that is encrypted with SK'_{hs-sp} and sends SK'_{hs-sp} to Hs.

Step 3. HS received with C, N_{sp}, V_{sp-hs} from SP computes $PUF_{hs}(C)$ and the R value and connects R and N_{hs} to hash them and create the session key, $SK_{hs\text{-}sp} = h(R' \mid\mid N_{hs})$. If the V_{sp-hs} received from HS is identical with the encrypted value with SK_{hs-sp} in connection with N_{sp} and N_{hs}, HS authenticates SP. In addition, the verification value, $V_{hs-sp} = D_{SKhs-sp}(C)$, is calculated by encrypting the C received from SP with SK_{hs-sp} and is sent to SP.

Step 4. SP received with V_{hs-sp} from HS encrypts C with SK'_{hs-sp}, and if $E_{SK'hs-sp}(C)$ is consistent with V_{hs-sp}, HS is authenticated. In addition, the used Challenge C and Response R' values are removed from the PUF'_{hs} DB.

Figure 5 shows the mutual authentication and key agreement phases between a middle-class sensor and a service provider, and between a middle-class sensor and a high-class sensor.

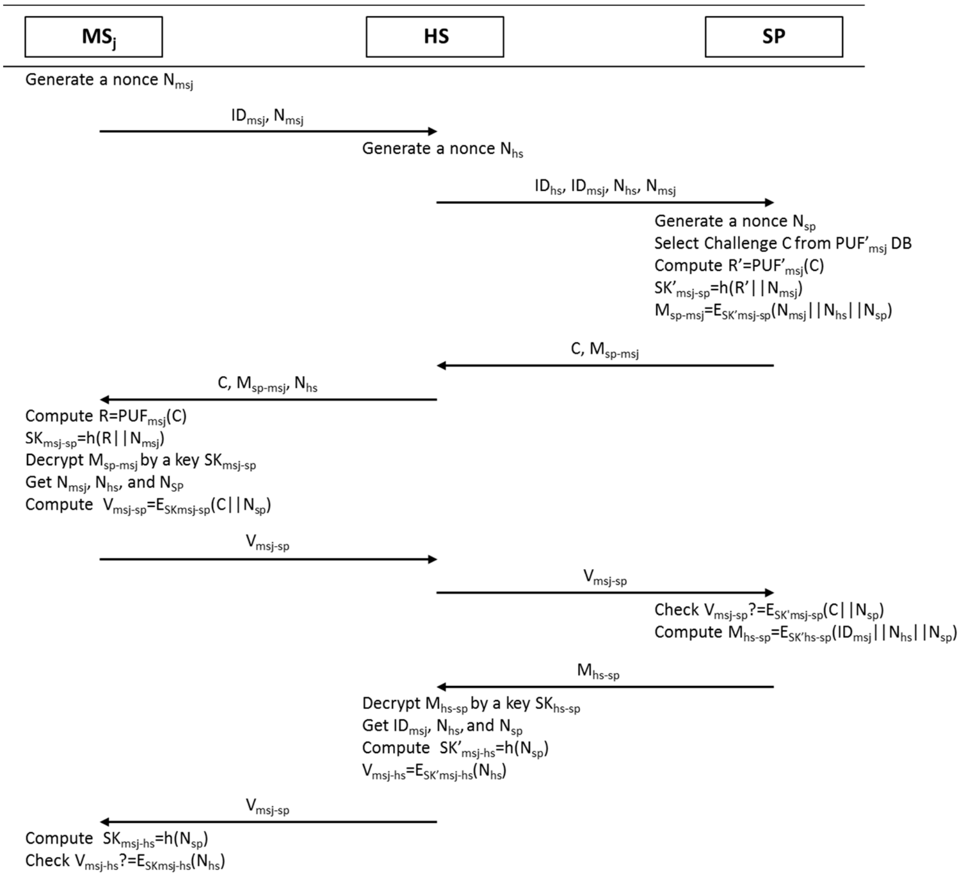
MS$_j$
HS
SP
Generate a nonce N$_{msj}$
ID$_{msj}$, N$_{msj}$
Generate a nonce N$_{hs}$
ID$_{hs}$, ID$_{msj}$, N$_{hs}$, N$_{msj}$
Generate a nonce N$_{sp}$
Select Challenge C from PUF'$_{msj}$ DB
Compute R'=PUF'$_{msj}$(C)
SK'$_{msj\text{-}sp}$=h(R'||N$_{msj}$)
M$_{sp\text{-}msj}$=E$_{SK'msj\text{-}sp}$(N$_{msj}$||N$_{hs}$||N$_{sp}$)
C, M$_{sp\text{-}msj}$
C, M$_{sp\text{-}msj}$, N$_{hs}$
Compute R=PUF$_{msj}$(C)
SK$_{msj\text{-}sp}$=h(R||N$_{msj}$)
Decrypt M$_{sp\text{-}msj}$ by a key SK$_{msj\text{-}sp}$
Get N$_{msj}$, N$_{hs}$, and N$_{SP}$
Compute V$_{msj\text{-}sp}$=E$_{SKmsj\text{-}sp}$(C||N$_{sp}$)
V$_{msj\text{-}sp}$
V$_{msj\text{-}sp}$
Check V$_{msj\text{-}sp}$?=E$_{SK'msj\text{-}sp}$(C||N$_{sp}$)
Compute M$_{hs\text{-}sp}$=E$_{SK'hs\text{-}sp}$(ID$_{msj}$||N$_{hs}$||N$_{sp}$)
M$_{hs\text{-}sp}$
Decrypt M$_{hs\text{-}sp}$ by a key SK$_{hs\text{-}sp}$
Get ID$_{msj}$, N$_{hs}$, and N$_{sp}$
Compute SK'$_{msj\text{-}hs}$=h(N$_{sp}$)
V$_{msj\text{-}hs}$=E$_{SK'msj\text{-}hs}$(N$_{hs}$)
V$_{msj\text{-}sp}$
Compute SK$_{msj\text{-}hs}$=h(N$_{sp}$)
Check V$_{msj\text{-}hs}$?=E$_{SKmsj\text{-}hs}$(N$_{hs}$)

Figure 5. Authentication and key agreement phase for a middle-class sensor.

Step 1. MS$_j$ generates a random number N_{msj} and sends it to HS with its identifier ID_{msj}.

Step 2. HS received with N_{msj} from MS$_j$ generates a random number N_{hs} and sends ID_{hs}, ID_{msj}, N_{hs}, and N_{msj} to SP.

Step 3. SP received with ID_{hs}, ID_{msj}, N_{hs}, and N_{msj} from HS generates a random number N_{sp} and selects Challenge C from the PUF'_{msj} DB. In addition, R' is computed from $PUF'_{msj}(C)$, which creates the session key SK'_{msj-sp} from $h(R' \mid\mid N_{msj})$. In addition, $(N_{msj} \mid\mid N_{hs} \mid\mid N_{sp})$ is encrypted with session key SK'_{msj-sp} and sends $M_{sp\text{-}msj} = E_{SK'msj-sp}(N_{msj} \mid\mid N_{hs} \mid\mid N_{sp})$ with C to HS.

Step 4. HS received with M_{sp-msj} from SP sends C, M_{sp-msj}, and N_{hs} to MS$_j$.

Step 5. MS$_j$ received with C, M_{sp-msj}, N_{hs} from HS computes $R = PUF_{msj}(C)$ and creates session key $SK_{msj\text{-}sp} = h(R \mid\mid N_{msj})$. N_{msj}, N_{hs}, and N_{sp} are acquired after decrypting M_{sp-msj} with session key SK_{msj-sp}, and if N_{hs}, which is received in plain text, is identical with N_{hs} acquired through decryption, HS and SP are authenticated. In addition, $V_{msj-sp} = E_{SKmsj-sp}(C||N_{sp})$ is created by encrypting $(C||N_{sp})$ with the session key SK_{msj-sp} and sends V_{msj-sp} to SP through HS.

Step 6. SP received with V_{msj-sp} from MS$_j$ through HS encrypts $(C||N_{sp})$ with the session key SK'_{msj-sp}. If $E_{SK'msj-sp}(C||N_{sp})$ is consistent with V_{msj-sp}, MS$_j$ is authenticated. In addition, $(ID_{msj}||N_{hs}||N_{sp})$ is encrypted with the session key, SK'_{hs-sp} and sends $M_{hs-sp} = E_{SK'hs-sp}(ID_{msj}||N_{hs})$ to HS. The used Challenge C and Response R' are removed from PUF'_{msj} *DB*.

Step 7. HS received with M_{hs-sp} from SP decrypts M_{hs-sp} with session key SK_{hs-sp}, and acquires ID_{msj}, N_{hs}, and N_{sp} and authenticates MS$_j$. In addition, N_{sp} acquired by decrypting the M_{hs-sp} is hashed, which creates the session key $SK'_{msj-hs} = h(N_{sp})$. N_{msj} is encrypted with the session key SK'_{msj-hs}, while creating $V_{msj-hs} = E_{SK'msj-hs}(N_{hs})$. The created $V_{msj\text{-}sp}$ is sent to MS$_j$.

Step 8. MS$_j$ received with V_{msj-sp} from HS hashes N_{sp} and creates the session key, SK_{msj-hs} and encrypts N_{hs} with the session key SK_{msj-hs} to see if it is consistent with V_{msj-hs}.

Figure 6 shows the mutual authentication and key agreement phases between a low-class sensor and service provider, between a low-class sensor and middle-class sensor, and between a low-class sensor and high-class sensor.

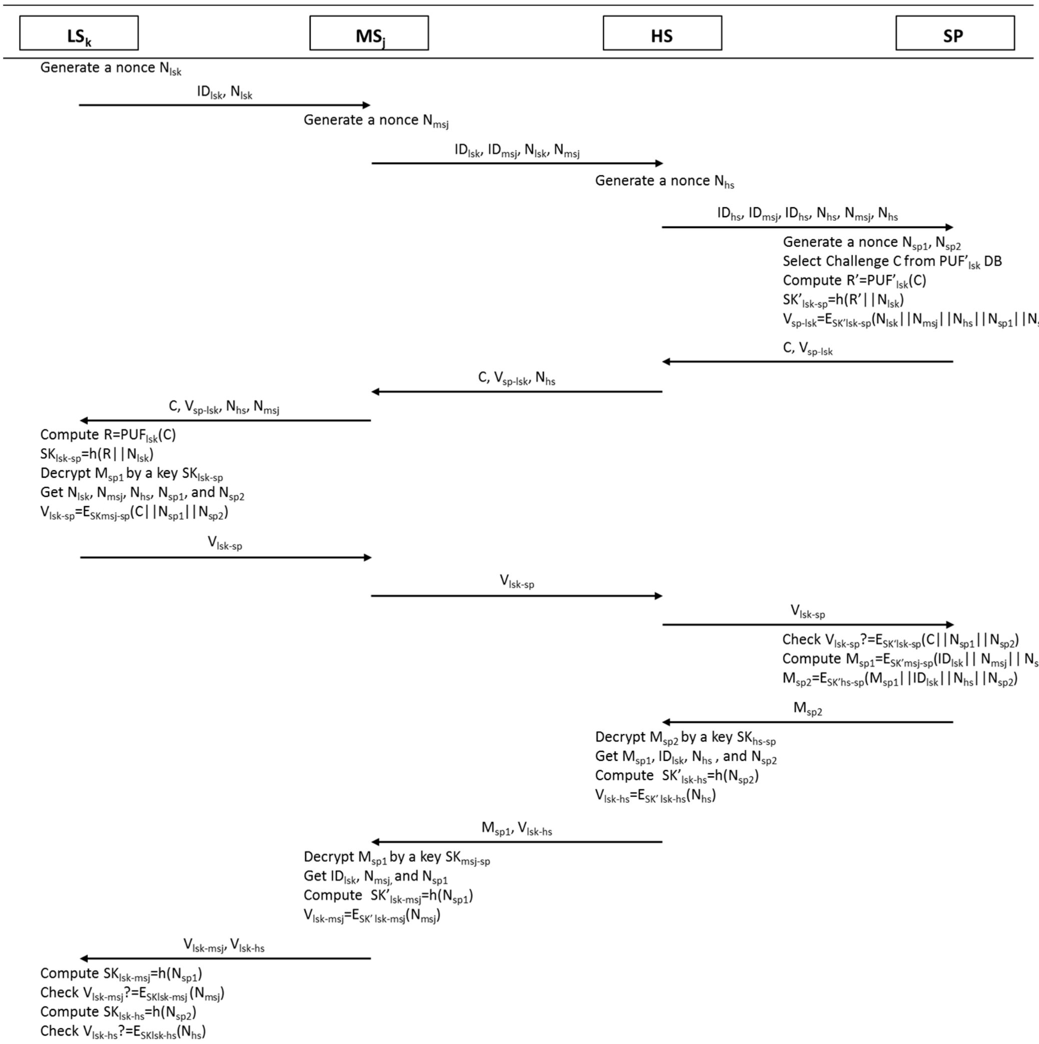
LS$_k$
MS$_j$
HS
SP
Generate a nonce N_{lsk}
ID_{lsk}, N_{lsk}
Generate a nonce N_{msj}
ID_{lsk}, ID_{msj}, N_{lsk}, N_{msj}
Generate a nonce N_{hs}
ID_{hs}, ID_{msj}, ID_{hs}, N_{hs}, N_{msj}, N_{hs}
Generate a nonce N_{sp1}, N_{sp2}
Select Challenge C from PUF'_{lsk} DB
Compute $R'=PUF'_{lsk}(C)$
$SK'_{lsk-sp}=h(R'||N_{lsk})$
$V_{sp-lsk}=E_{SK'lsk-sp}(N_{lsk}||N_{msj}||N_{hs}||N_{sp1}||N_{sp2})$
C, V_{sp-lsk}
C, V_{sp-lsk}, N_{hs}
C, V_{sp-lsk}, N_{hs}, N_{msj}
Compute $R=PUF_{lsk}(C)$
$SK_{lsk-sp}=h(R||N_{lsk})$
Decrypt M_{sp1} by a key SK_{lsk-sp}
Get N_{lsk}, N_{msj}, N_{hs}, N_{sp1}, and N_{sp2}
$V_{lsk-sp}=E_{SKmsj-sp}(C||N_{sp1}||N_{sp2})$
V_{lsk-sp}
V_{lsk-sp}
V_{lsk-sp}
Check $V_{lsk-sp}?=E_{SK'lsk-sp}(C||N_{sp1}||N_{sp2})$
Compute $M_{sp1}=E_{SK'msj-sp}(ID_{lsk}||N_{msj}||N_{sp1})$
$M_{sp2}=E_{SK'hs-sp}(M_{sp1}||ID_{lsk}||N_{hs}||N_{sp2})$
M_{sp2}
Decrypt M_{sp2} by a key SK_{hs-sp}
Get M_{sp1}, ID_{lsk}, N_{hs} , and N_{sp2}
Compute $SK'_{lsk-hs}=h(N_{sp2})$
$V_{lsk-hs}=E_{SK'lsk-hs}(N_{hs})$
M_{sp1}, V_{lsk-hs}
Decrypt M_{sp1} by a key SK_{msj-sp}
Get ID_{lsk}, $N_{msj,}$ and N_{sp1}
Compute $SK'_{lsk-msj}=h(N_{sp1})$
$V_{lsk-msj}=E_{SK'lsk-msj}(N_{msj})$
$V_{lsk-msj}$, V_{lsk-hs}
Compute $SK_{lsk-msj}=h(N_{sp1})$
Check $V_{lsk-msj}?=E_{SKlsk-msj}(N_{msj})$
Compute $SK_{lsk-hs}=h(N_{sp2})$
Check $V_{lsk-hs}?=E_{SKlsk-hs}(N_{hs})$

Figure 6. Authentication and key agreement phase for a low-class sensor.

Step 1. LS$_k$ generates the random number N_{lsk} and sends ID_{lsk}, and N_{lsk} to MS$_j$.

Step 2. MS$_j$ received with ID_{lsk}, N_{lsk} from LS$_k$ generates the random number N_{msj} and sends ID_{lsk}, ID_{msj}, N_{lsk}, and N_{msj} to HS.

Step 3. HS received with ID_{lsk}, ID_{msj}, N_{lsk}, and N_{msj} from MS$_j$ generates the random number N_{hs}, and sends ID_{hs}, ID_{msj}, ID_{hs}, N_{hs}, N_{msj}, and N_{hs} to SP.

Step 4. SP received with ID_{hs}, ID_{msj}, ID_{hs}, N_{hs}, N_{msj}, and N_{hs} from HS generates the random numbers N_{sp1}, N_{sp2}. In addition, it selects Challenge C from the PUF'_{lsk} DB by computing $R' = PUF'_{lsk}(C)$ and producing the session key $SK'_{lsk-sp} = h(R'||N_{lsk})$. In addition, $(N_{lsk}||N_{msj}||N_{hs}||N_{sp1}||N_{sp2})$ is encrypted with session key SK'_{lsk-sp} and sends C and V_{sp-lsk} to HS.

Step 5. HS received with C, V_{sp-lsk} from SP sends C, V_{sp-lsk}, and N_{hs} to MS$_j$.

Step 6. MS$_j$ received with C, V_{sp-lsk}, N_{hs} from HS sends C, V_{sp-lsk}, N_{hs}, and N_{msj} to LS$_k$.

Step 7. LS$_k$ received with C, V_{sp-lsk}, N_{hs}, and N_{msj} from MS$_j$ hashes R and N_{lsk} acquired by computing $R = PUF_{lsk}(C)$ and creates the session key $SK_{lsk-sp} = h(R||N_{lsk})$. In addition, M_{sp1} is decrypted by SK_{lsk-sp} while acquiring N_{lsk}, N_{msj}, N_{hs}, N_{sp1}, and N_{sp2}. If N_{hs} and N_{msj} acquired by decrypting N_{hs}, N_{msj} and M_{sp1} are identical, MS, HS, and SP are authenticated. $(C||N_{sp1}||N_{sp2})$ is encrypted with the session key $SK_{lsk\text{-}sp}$, which creates the $V_{lsk\text{-}sp}$=$E_{SKmsj-sp}(C||N_{sp1}||N_{sp2})$ and sends V_{lsk-sp} to SP through MS$_j$ and HS.

Step 8. SP received with V_{lsk-sp} through MS$_j$ and HS from LS$_k$ encrypts $(C||N_{sp1}||N_{sp2})$ with SK'_{lsk-sp} and authenticates LS$_k$ if $E_{SK'lsk-sp}(C||N_{sp1}||N_{sp2})$ and V_{lsk-sp} are identical. In addition, $(ID_{lsk}||N_{msj}||N_{sp1})$ is encrypted with the session key SK'_{hs-sp}, creating $M_{sp1} = E_{SK'msj-sp}(ID_{lsk}||N_{msj}||N_{sp1})$, and $(M_{sp1}||ID_{lsk}||N_{hs}||N_{sp2})$ is encrypted with the session key SK'_{hs-sp} creating $M_{sp2} = E_{SK'hs-sp}(M_{sp1}||ID_{lsk}||N_{hs}||N_{sp2})$ and sending M_{sp2} to HS. The used Challenge C and Response R' are removed from the PUF'_{msj} DB.

Step 9. HS received with M_{sp2} from SP decrypts M_{sp2} with the session key, SK_{hs-sp}, and acquires M_{sp1}, ID_{lsk}, N_{hs}, and N_{sp2} and authenticates LS$_k$. In addition, N_{sp2} is hashed creating $SK'_{lsk-hs} = h(N_{sp2})$ and encrypting the N_{hs} with session key SK'_{lsk-hs}, while creating $V_{lsk\text{-}hs}$=$E_{SK'lsk-hs}(N_{hs})$. M_{sp1}, V_{lsk-hs} are received by MS$_j$.

Step 10. MS$_j$ received with M_{sp1}, V_{lsk-hs} from HS decrypts M_{sp1} with the session key SK_{msj-sp}, and acquires ID_{lsk}, N_{msj}, and N_{sp1} and authenticates LS$_k$. In addition, N_{sp1} is hashed, which creates the session key $SK'_{lsk-msj} = h(N_{sp1})$, encrypts N_{msj} with the session key $SK'_{lsk-msj}$, and creates $V_{lsk-msj} = E_{SK'lsk-hs}(N_{msj})$. $V_{lsk-msj}$ and V_{lsk-hs} are received by MS$_j$.

Step 11. LS$_k$ received with $V_{lsk-msj}$, V_{lsk-hs} from MS$_j$ hashes N_{sp1} creating the session key $SK_{lsk-msj} = h(N_{sp1})$, encrypts N_{msj} with the session key $SK_{lsk-msj}$, and encrypts N_{hs} with the session key SK_{lsk-hs} to see if it is identical with V_{lsk-hs}.

4. Security and Performance Analysis

4.1. Security Analysis

Suggested techniques have classified each of the sensors into three classes depending on the capabilities of computing ability and battery life, while constituting a hierarchical topology and performing the mutual authentication and key agreement phases amongst sensors and between the sensor and service provider. In addition, our suggested techniques are secure against keys being leaked, forward secrecy, eavesdropping, and replay attacks by malicious attackers, and they are highly secure and efficient compared to other methods. Therefore, they support the row resource sensor environment.

4.1.1. Performance Analysis

Table 2 shows whether our proposed method, and other existing schemes, support the topology for ESNs, whether they are designed to address security while taking the low resources of the device with small computing power and battery life into account, and whether different sensor performances are also considered.

Table 2. Comparative performance analysis between smart home schemes.

Performance Requirements	Alessandro et al. [29]	Vijay et al. [30]	Debraj et al. [31]	Basma M et al. [32]	Proposed Scheme
Topology for smart home	X	X	X	X	O
Security	X	O	X	O	O
Different sensor performances	X	X	X	O	O
Low Resource	O	X	O	O	O

O: Supported; X: Not supported.

Topology for ESNs in Smart Homes: The suggested topology for ESNs in smart homes has been classified into three classes depending on the capability of the sensors to be installed in the home to constitute the hierarchical topology. Sensors in each layer tend to have a shorter communication distance and perform fewer computing calculations in descending order from high- to low-class. Therefore, it has solved the issue of inefficiency in previous sensor networks which only communicated with adjacent sensors without considering sensor capability, while making topology control easier by designating the targets that are to be communicated with in the sensors in each layer.

Different Sensor Performances: Due to the development of IoT technology, various sensors have been developed, and these sensors have been distributed and used in a smart home environment with diverse capabilities. In order to efficiently utilize the various capabilities of sensors, we classified our proposed authentication techniques into three layers depending on their capabilities. Sensors with lower resources were designed to perform fewer computing calculations and consume less memory space than those with higher resources. For example, low-class sensors are made up of one sensor with relatively high resources and a security channel. However, high-class sensors communicate with multiple sensors with low security resources by differentiating the distribution of calculations in each class.

Low Resources: Most of the sensors in a smart home tend to have a relatively low computing ability or power than existing computing devices. Therefore, security techniques used in the existing computing devices are not appropriate. Our technique considers the low resources of sensors by utilizing PUFs for establishing mutual authentication and a security channel. In addition, a secret value needed to establish a security channel was minimally required for small storage space in a sensor.

4.1.2. Security Analysis

Table 3 shows how secure our proposed method and other existing schemes are against various security threats.

Table 3. Comparative security analysis between sensor schemes.

Threates	ELK [22]	LKH [23]	CoGKTK [24]	sGIM [25]	Proposed Scheme
Leaked key	Not-support	Not-support	Support	Support	Support
Forward Secrecy	X	X	O	X	O
Mutual Authentication	X	X	O	X	O
Eavesdropping	X	X	X	X	O
Replay Attack	X	X	O	O	O

O: Secure; X: Vulnerable.

Leaked Key: Sensors in each class share a session key with service providers and the sensors of other classes in order to establish a security channel with the service provider. When each sensor shares a session key with a service provider, the Response R value of the PUFs registered in the service provider in advance is used. PUFs have a unique challenge-response value. Therefore, it is not possible to infer a session key between a sensor and service provider. When sharing a session key among sensors in other classes, each sensor shares the secret value for creating a session key through a security channel that has been established with the service provider. This makes it possible to prevent malicious attackers from hacking the key.

Forward Secrecy: Malicious attackers might attempt to steal the current session key used in the communication between the sensor and service provider or amongst sensors and restore the information exchanged in the past by inferring previous session keys. However, sensors in each class establish a security channel with the service provider. The session key used at this time utilizes a unique response value of PUFs that is not re-used. Therefore, it is not possible to infer a past session key with the current session key. In addition, since the current session key amongst sensors is created by a secret value generated randomly by the service provider, it is not possible to infer a past session key even if the current session key is stolen.

Mutual Authentication: Each sensor in the smart home environment is required to establish a security channel through mutual authentication to securely exchange information. Our technique uses PUFs in each sensor to perform mutual authentication with the service provider and, through the service provider, in the communication amongst the three suggested layer-based classes. Therefore, it is possible to establish a security channel.

Eavesdropping: Malicious attackers eavesdrop on the information exchanged between a smart home sensor and service provider or amongst sensors in the smart home. They steal sensitive information or use it for malicious purposes. With our proposed technique, the information exchanged in plain text only includes the Challenge C value that is not re-used, the ID of the sensor and the service provider, and the random number created by each sensor and service provider. Therefore, malicious attackers cannot steal important information through eavesdropping. In addition, our technique is secure against forward secrecy and keys being leaked. Therefore, it is not possible to acquire information from an encrypted message.

Replay Attack: Information exchanged in plain text amongst sensors, or between the sensor and service provider, can be stolen by malicious attackers and used for a replay attack. However, our technique uses the challenge-response system of PUFs that cannot be re-used. Therefore, it is not possible to re-use an authentication message between the sensor and service provider. In addition, an authentication message among devices in each class uses a random value, which makes a replay attack impossible.

4.2. Computing Resource Analysis

Table 4 provides the computing resource analysis when low-class sensors, middle-class sensors, a high-class sensor, and a service provider were applied in the provisioning phase and authentication and key agreement phases in our proposed technique. It is assumed that the number of low-class sensors is K, the number of middle-class sensors is J, and the number of sensors connected to each middle-class sensors is k (k < K). The number of low-class sensors, K, was assumed to be greater than the number of middle-class sensors, J (J < K). The low-class sensor with the lowest computing power in the classes and SP do not perform complicated computations. They only compute PUFs() n times in the initial registration procedures the most. A middle-class sensor with mid-computing power only performs the calculation k times more than a low-class sensor, except for PUFs() computation. In addition, a high-class sensor and service provider are equipped with enough computing resources and perform more calculations than low- and middle-class sensors. In addition, the most complicated calculation is for decryption in the use of the matching key, which is mostly by the high-class sensor and SP. Calculation was dispersed in the order of low-class sensor, middle-class sensor, high-class sensor, and SP depending on computing resources. Mutual authentication and the security channel were established with the minimum number of calculations.

Table 4. Comparative computing resource analysis between communication objects.

Calculation Item	LS	MS	HS	SP
PUF	$n + 1$	$n + 1$	$n + 1$	$(n + 1)(1 + J + K)$
Hash	2	2 + k	1 + J + K	1 + J + K
Encryption	3	2 + k	2 + J + K	2 + 3J + 4K
Decryption	1	1 + k	J + K	-
Nonce generation	1	1 + k	1 + J + K	1 + J + 2K

4.3. Storage Resource Analysis

Table 5 provides the analysis of storage resources, which are required in the provisioning phase and authentication and key agreement phase in our technique. Each of the sensors and service providers have DI, Challenge C, Response R, session key (SK), message (M), verification value (V), and a random number (N) for each phase. A low-class sensor with the lowest storage capacity has the smallest ID as it possesses its own ID, higher-class IDs, and the IDs of the SP. High-class sensors have the additional IDs of lower-class sensors that require a greater storage capacity. SP has the IDs of all the sensors. Sensors in each class, except for SP, compute PUFs() for mutual authentication by saving one Challenge C and one Response R. SP possesses Challenge C and Response R in all the sensors. SK is required in each communication interval. Therefore, low-class sensors only save the three session keys needed for communication with high-class sensors. High-class sensors and service providers possess the additional session keys needed for communication with low-class sensors. High-class sensors and service providers store more of the message (M), verification value (V), and random number (N) due to there being more targets to communicate with than low-class sensors. Storage resources were allocated in the order of low-class sensor, middle-class sensor, high-class sensor, and SP, depending on the capacity of the storage resource. Low-class sensors with the smallest amount of storage resources are equipped with the minimum amount of storage capacity.

Table 5. Comparative storage resource analysis between communication objects.

Storage Item	LS	MS	HS	SP
ID	4	3 + k	3 + K	2 + J + K
Challenge C	1	1	1	$n + Jn + \mathrm{kN}$
Response R	1	1	1	$n + Jn + \mathrm{kN}$
SK	3	2 + k	1 + J + K	1 + J + K
M	1	1 + k	J + 2K	2(J + K)
V	3	2 + k	2 + J + K	2 + J + 2K
N	3	3 + k	2 + 3J + 3K	2 + 3J + 5K

5. Conclusions

A smart home is a form of technology that can collect and analyze the information of those who reside there by using various sensors and emerging technologies. Emerging technologies have been combined with IoT, which has resulted in creating smart home service as a new field of ESNs. As such, various companies and research institutions around the world have been proceeding with research and development. Most of the sensors in this service are equipped with low-power and low-computing ability. Therefore, it is very important to deliver sensing information without placing a burden on the sensor, which is why topology control is required. In this study, sensors with low resources and sensors with diverse capabilities operating in the smart home were considered and classified into low-, middle-, and high-class, depending on their ability to constitute the hierarchical topology. Therefore, in this study, we have proposed a technique for ensuring secure communication, and consuming low computing and storage resources with PUF while efficiently utilizing the abilities of the sensors. In addition, our technique has been evaluated to be secure against various security threats via the execution of a security analysis. Our proposed scheme was also evaluated by analyzing the computing resources and storage resources needed by each communicator. However, there are still some problems, such as multi-platform compatibility and security policies set according to the importance of information, that we will study in the future. We believe that our scheme will be able to establish secure communication with fewer resources in the smart home sensor network.

Author Contributions: Mansik Kim designed the protocol; Jungsuk Song researched the related work; Mansik Kim and Kyung-Soo Lim performed and analysed the data; and Moon-seog Jun and Mansik Kim wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Whitmore, A.; Anurag, A.; Li, D.X. The Internet of Things? A survey of topics and trends. *Inf. Syst. Front.* **2015**, *17*, 261–274. [CrossRef]
2. Mainetti, L.; Patrono, L.; Vilei, A. Evolution of wireless sensor networks towards the internet of things: A survey. In Proceedings of the 2011 19th International Conference on Software, Telecommunications and Computer Networks (SoftCOM), Split, Croatia, 15–17 September 2011.
3. Kun, P. A secure network for mobile wireless service. *J. Inf. Process. Syst.* **2013**, *9*, 247–258.
4. Bayram, I.S.; Papapanagiotou, I. A survey on communication technologies and requirements for internet of electric vehicles. *EURASIP J. Wirel. Commun. Netw.* **2014**, *2014*, 1–18. [CrossRef]
5. Hashizume, K.; Rosado, D.G.; Fernández-Medina, E.; Fernandez, E.B. An analysis of security issues for cloud computing. *J. Internet Serv. Appl.* **2013**, *4*, 5. [CrossRef]
6. Ablondi, W. *2014 Smart Home Systems and Services Forecast Global Total*; Strategy Analytics: Boston, MA, USA, 2014.
7. Zhang, F.; Xu, Y.; Chou, J. A novel petri nets-based modeling method for the interaction between the sensor and the geographic environment in emerging sensor networks. *Sensors* **2016**, *16*, 1571. [CrossRef] [PubMed]
8. Bays, L.R.; Oliveira, R.R.; Barcellos, M.P.; Gaspary, L.P.; Madeira, E.R.M. Virtual network security: Threats, countermeasures, and challenges. *J. Internet Serv. Appl.* **2015**, *6*, 1. [CrossRef]
9. Joo, J.W.; Lee, J.K.; Park, J.H. Security considerations for a connected car. *J. Converg.* **2015**, *6*, 1–9.
10. Ahn, H.; Kim, H.; Park, J.R. Smart monitoring of indoor asbestos based on the distinct optical properties of asbestos from particulate matters. *J. Converg.* **2014**, *5*, 22–25.
11. Li, D.X.; He, W.; Li, S. Internet of things in industries: A survey. *IEEE Trans. Ind. Inform.* **2014**, *10*, 2233–2243.
12. Hewlett Packard Enterprise. *How Safe Are Home Security Systems? An HPE Study on IoT Security*; HPE: Palo Alto, CA, USA, 2015.
13. Kang, J.; Han, J.; Park, J.H. Design of IP camera access control protocol by utilizing hierarchical group key. *Symmetry* **2015**, *7*, 1567–1586. [CrossRef]
14. Im, H.; Kang, J.; Park, J.H. Certificateless based public key infrastructure using a DNSSEC. *J. Converg.* **2015**, *6*, 26–33.
15. Kang, J.; Kim, M.; Park, J.H. A reliable TTP-based infrastructure with low sensor resource consumption for the smart home multi-platform. *Sensors* **2016**, *16*, 1036. [CrossRef] [PubMed]
16. Kwon, T.; Lee, J.; Choi, H.; Yi, O.; Ju, S. Efficiency of LEA compared with AES. *J. Converg.* **2015**, *6*, 16–25.
17. Ng, C.K.; Wu, C.H.; Ip, W.H.; Zhang, J.; Ho, G.T.S.; Chan, C.Y. Network topology management optimization of wireless sensor network (WSN). In Proceedings of the International Conference on Intelligent Computing, Lanzhou, China, 2–5 August 2016; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2016; pp. 850–859.
18. Sohrabi, K.; Gao, J.; Ailawadhi, V.; Pottie, G.J. Protocols for self-organization of a wireless sensor network. *IEEE Pers. Commun.* **2000**, *7*, 16–27. [CrossRef]
19. Shnayder, V.; Hempstead, M.; Chen, B.R.; Allen, G.W.; Welsh, M. Simulating the power consumption of large-scale sensor network applications. In Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, Baltimore, MD, USA, 3–5 November 2004.
20. Gaur, M.S.; Pant, B. Trusted and secure clustering in mobile pervasive environment. *Hum.-Centric Comput. Inf. Sci.* **2015**, *5*, 32. [CrossRef]

21. Kang, J.; Park, J.H.; Suk, S. Design of a distributed personal information access control scheme for secure integrated payment in NFC. *Symmetry* **2015**, *7*, 935–948. [CrossRef]
22. Penrig, A.; Song, D.; Tygar, D. Elk, a new protocol for efficient large-group key distribution. In Proceedings of the 2001 IEEE Symposium on Security and Privacy (S & P 2001), Oakland, CA, USA, 14–16 May 2000.
23. Wong, C.K.; Mohamed, W.; Simon, G.; Lam, S. *Secure Group Communications Using Key Graphs*; Tech. Rep.; ACM SIGCOMM Computer Communication Review: Waterloo, ON, Canada, 1998.
24. Nguyen, T.-D.; Huh, E.-N. An efficient Key management for secure multicast in Sensor-Cloud. In Proceedings of the 2011 First ACIS/JNU International Conference, Jeju Island, Korea, 23–25 May 2011.
25. Hassan, M.M.; Song, B.; Huh, E.-N. A framework of sensor-cloud integration opportunities and challenges. In Proceedings of the 3rd International Conference on Ubiquitous Information Management and Communication, Suwon, Korea, 15–16 January 2009.
26. Gupta, G.P.; Misra, M.; Garg, K. An energy efficient distributed approach-based agent migration scheme for data aggregation in wireless sensor networks. *JIPS* **2015**, *11*, 148–164.
27. Hwang, K.-I.; Jang, I. Ultra low power data aggregation for request oriented sensor networks. *JIPS* **2014**, *10*, 412–428. [CrossRef]
28. Dahane, A.; Berrached, N.-E.; Loukil, A. A virtual laboratory to practice mobile wireless sensor networks: A case study on energy efficient and safe weighted clustering algorithm. *J. Inf. Process. Syst.* **2015**, *11*, 205–228.
29. Viani, F.; Robol, F.; Polo, A.; Massa, A. Wireless architectures for heterogeneous sensing in smart home applications: Concepts and real implementation. *IEEE Proc.* **2013**, *101*, 2381–2396. [CrossRef]
30. Sivaraman, V.; Gharakheili, H.H.; Vishwanath, A.; Boreli, R.; Mehani, O. Network-level security and privacy control for smart-home IoT devices. In Proceedings of the 2015 IEEE 11th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), Avignon, France, 19–21 October 2015.
31. Basu, D.; Moretti, G.; Gupta, G.S.; Marsland, S. Wireless sensor network based smart home: Sensor selection, deployment and monitoring. In Proceedings of the 2013 IEEE Sensors Applications Symposium (SAS), Galveston, TX, USA, 19–21 February 2013.
32. El-Basioni, B.M.M.; El-kader, S.M.A.; Abdelmonim, M. Smart home design using wireless sensor network and biometric technologies. *Int. J. Appl. Innov. Eng. Manag.* **2013**, *2*, 413–429.
33. Gassend, B.; Clarke, D.; van Dijk, M.; Devadas, S. Silicon physical random functions. In Proceedings of the 9th ACM conference on Computer and communications security, Washington, DC, USA, 18–22 November 2002.
34. Gassend, B.L.P. Physical Random Functions. Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2003.
35. Bielefeldt, J.; Chellappan, S. Sensor authentication in collaborating sensor networks. In Proceedings of the 2014 13th Annual Mediterranean Ad Hoc Networking Workshop (MED-HOC-NET), Los Angeles, CA, USA, 2–4 June 2014.
36. Delvaux, J.; Peeters, R.; Gu, D.; Verbauwhede, I. A survey on lightweight entity authentication with strong PUFs. *ACM Comput. Surv. (CSUR)* **2015**, *48*, 26. [CrossRef]