Anonymous Secure Framework in Connected Smart Home Environments

Pardeep Kumar, *Member, IEEE,* An Braeken, Andrei Gurtov, *Senior Member, IEEE,* Jari Iinatti, *Senior Member, IEEE,* and Phuong Hoai Ha,

***Abstract*—The smart home is an environment where heterogeneous electronic devices and appliances are networked together to provide smart services in an ubiquitous manner to the individuals. As the homes become smarter, more complex and technology dependent, the need for an adequate security mechanism with minimum individual's intervention is growing. The recent serious security attacks have shown how the Internet-enabled smart homes can be turned into very dangerous spots for various ill intentions, and thus lead the privacy concerns for the individuals. For instance, an eavesdropper is able to derive the identity of a particular device/appliance via public channels that can be used to infer in the life pattern of an individual within the home area network. This paper proposes an anonymous secure framework (ASF) in connected smart home environments, using solely lightweight operations. The proposed framework in this paper provides efficient authentication and key agreement, and enables devices (identity and data) anonymity and unlinkability. One-time session key progression regularly renews the session key for the smart devices and dilutes the risk of using a compromised session key in the ASF. It is demonstrated that computation complexity of the proposed framework is low as compared to the existing schemes, while security has been significantly improved.**

***Index Terms*—Smart home, Internet of things, Anonymity, Key agreement, Unlinkability.**

I. INTRODUCTION

SMART home is a technological advancement and concept for monitoring and controlling home appliances through intelligent and coordinated networks and technologies. Smart spaces consist of a plethora of heterogeneous devices, for instance, multiple cameras, microphones, sensors, actuators, smart appliances, smart curtains and so on. Such a development has been leading individuals to the new era of technology, and the era of the Internet of things (hereafter IoT) where all the appliances and devices are getting tiny and controllable via the Internet, thus enabling people to enjoy network based services, such as home climate control, energy management, video on demand, music on demand, remote healthcare, e-commerce, remote control, and other similar services [1] [2]. Moreover, the number of smart systems will dramatically increase as the consumer IoT continues to evolve. As a consequence, the individual will become more and more dependent on smart systems.

However, the internal network of a smart environment consists of a number of different communication and network technologies. Examples of some popular standards and protocols related to home automation include X10, UPB, INSTEON, Z-Wave, and ZigBee [3]. X10 and UPB utilize existing electricity, or a power-line network. INSTEON is a dual-band mesh network topology employing AC-power lines and a radio-frequency protocol to communicate with devices. Particularly, Z-Wave [4] and ZigBee [5] are mostly utilized technologies, having low-power wireless communication capability. ZigBee supports a specific home automation profile, and Z-wave is optimized for the reliable low-latency communication of small data packets with data rates of up to 100 Kbps [6].

As a technological convergence, many of the home devices or appliances are always connected to the Internet over wireless communications, within the home area network (HAN). Connecting smart home appliances to wireless networks and to the Internet, however, makes individuals vulnerable to malicious attacks. If the smart devices (e.g., smart lights, appliances, smart watches, smart meters, smart fridge and many more) within a smart home are inadequately networked, that will open the occupant of smart home up to much wider range of security threats including identity theft, device counterfeiting, etc. In January 2014, it was, for instance, discovered that more than 750,000 consumer devices including home routers, televisions, fridges, thermostats, smart locks, televisions, and so on, had been compromised and/or spied on the individual [7]. Another research revealed 250 different security flaws, which equates to 25 vulnerabilities per smart device [8]. This is due to the weak security design of the proprietary technologies, and lack of capable security standards of smart objects [9].

Moreover, the fine-grained data of smart devices (e.g., smart lights, locks, thermostat, climate sensors, appliances, windows, smart television, smart meter information, etc.) are transmitted via insecure wireless channels in a HAN. Such sensitive information may be concealed, controlled and linked without users' consent. For instance, a non-intrusive load monitoring (NILM) algorithm could gather home appliances information and identification (e.g., device identity) from load profiles [10] [11]. In other words, an unauthorized user could take advantage of NILM algorithm to analyse networked devices

P. Kumar is with the Department of Computer Science, University of Oxford, OX1 3QD, United Kingdom (e-mails: pardeep.kumar@cs.ox.ac.uk)

A. Braeken is with Industrial Engineering INDI, Vrije Universiteit Brussel, Belgium (e-mail: an.braeken@vub.ac.be)

A. Gurtov is with Linkoping University, Sweden and ITMO University, Russia (e-mail (gurtov@acm.org))

J. Iinatti is with University of Oulu, P. O. Box 4500, FI-90014, Finland (e-mails: ji@ee.oulu.fi)

P. H. Ha is with the Department of Computer Science, UiT The Arctic University of Norway, N-9037, Tromso, Norway (e-mails: phuong.hoai.ha@uit.no)

within the home, and hence detect and/or link the individual life patterns, daily routines, and habits for the sake of profit, theft and advertisements. This raises, therefore, two main security concerns: (i) how to network the smart devices/appliances within a HAN without being identifiable (i.e., anonymity), and (ii) how to network the smart devices/appliances without being able to distinguish relationships between two devices (i.e., unlinkability) in smart home over the public network.

Towards the smart home security, related work mainly focuses on the device authentication in smart home use-cases [12] [13] [14] [15] [16] [17] [18]. However, most of the proposed schemes incurred a high amount of overhead for authentication [13] [14] [15] [16] [17]. None of the schemes are considering anonymity and unlinkability in the smart homes where the malicious attacker can easily disclose (and link) the appliances and devices' identities by utilizing the NILM algorithm [10] [11], and so breach the home network security and privacy. It therefore becomes necessary to design an efficient security framework in connected smart homes to realize its security protection (considering anonymity and unlinkability) for the individuals.

In the design of a secure framework for the smart home environments, providing mutual authentication and key agreement are the required first steps to prevent illegal use of home appliances and systems. Besides a secure and efficient authentication, the security framework should satisfy the following merits: (i) **Anonymity and unlinkability**: Hide of appliance identity, sensor presence and data-collection activity from unauthorized tracking. Even a malicious device should not be able to reveal the identity and relationship of devices communicating within the home network. (ii) **Authentication and integrity**: The source of the information can be corroborated and it is ensured that the protocol information has not been altered by unauthorized or unknown means. (iii) **Low communication cost and computation complexity**: Usually, a battery-powered smart device generally has severe resource constraints on its ability to process and communicate data. As a result, the secure framework must take communication and computation efficiency into consideration. (iv) **Security safeguard**: The secure framework should have ability to resist possible attacks (e.g., replay attack, impersonation attack) such that it can be applied in the real home environments.

Considering the above mentioned security merits, we design and implement an anonymous secure framework (ASF) for the smart home environments. In the proposed ASF, the smart devices/appliances can communicate with the home gateway in a HAN, while providing the above mentioned security services. The main contributions are in three-fold, as follows.

- First, we present a novel ASF scheme that is very lightweight and efficient, reducing significantly computation and communication cost. To the best of our knowledge, the new ASF scheme is the first scheme that considers the anonymity and unlinkability in the smart homes. Inspired by the fact of smart home use-cases, which are of very sensitive and multidimensional nature, the ASF scheme utilizes hashing and symmetric cryptosystems to achieve device anonymity, efficient authentication and key agreement between two communicating devices within the home area network. Compared with the existing schemes, it leads to significantly reduced computation and communication cost.
- Second, we conduct simulation for formal security analysis of the security strength and anonymity of the new ASF scheme. In particular, we use AVISPA (automated verification of Internet security protocol and application) tool that has been widely used by the standardisation bodies (e.g., Internet Engineering Task Force (IETF)), and by the academic research to verify security of the protocols (e.g., [18], [19]). In addition, we use BAN logic to formally verify that the smart devices within the HAN are semantically achieve the security goals.
- Finally, we conduct comparative performance analysis of the new ASF scheme, showing that the proposed ASF requires indeed lower computational and communicational costs than [17] [20].

The rest of this paper is organized as follows. Section II reviews the related work in smart home use-cases security. Section III presents the system model, assumptions, and notations, and Section IV presents the proposed anonymous secure framework (ASF). Section V introduces security analysis based on the AVISPA tool, the BAN-logic and informal analysis, and Section VI discusses performance analysis. Finally, in Section VII we draw the conclusion.

II. Related work

Hoang-Pishva suggested a TOR-based anonymous communication approach to secure smart home appliances in [12]. Usually the Internet users use TOR as an Internet browser, which operates as an anonymous browser where only those surfing activities done within the browser are anonymized, but authentication is not being performed. Moreover, the scheme utilizes public-key cryptography, which is quite expensive for resource hungry devices. Vaidya et al. [17] proposed a device authentication mechanism for smart energy home area networks. Based on elliptic curve cryptography (ECC), each device obtains an implicit certificate from the certificate authority. The mutual authentication is being performed and a session key is established between two involved entities, where devices' identities are being used as a plain-text. Authors claimed their scheme is efficient compared to other existing schemes. However, security analysis did not provide details.

Kumar et al [18] introduced lightweight and secure session key establishment scheme for smart home environments. A short authentication token is used to verify the legitimacy of the smart devices. Authors claimed that the scheme is secure against various popular attacks, such as denial-of-service and eavesdropping attacks. However, in [18], the home gateway is required to store the smart device secret keys in a table and anonymity and unlinkability are not considered. Santoso-Vun [20] suggested a strong security in IoT for smart home systems considering user convenience in operating the system. The protocol uses ECC due to its high security level per key size, while the use of pre-shared secret keys (K) removes the need to establish additional public key infrastructure for the system. After the authentication process is done, both parties

(i.e., sender and receiver) can use the Elliptic Curve Diffie Hellman (ECDH) primitive to create a shared key for the subsequent symmetric encryption.

In [21], Ayday-Rajagopal noticed that the existing HAN technologies, for instance, ZigBee, Z-wave, and INSTEON support security only up to a certain level. For the smart grid-enabled HAN, authors introduced three different secure device authentication mechanisms: (i) authentication mechanism between the gateway and the smart meter; (ii) authentication between the smart appliances and the HAN; and (iii) authentication between the transient devices and the HAN. To execute authentication, the schemes are depending on Internet service provider [21]. Logue et al [22], proposed a multi-tiered authentication method for facilitating secure communication amongst smart home devices and cloud-based server. The scheme exploits the client-server architecture where the remote server may provide or refuse access to the client device based on a level of authentication of the client device. Here, level of authentication means the client device may authenticate its identity using different device credentials or other characteristics/relationships. For details, the reader may refer to [22].

A dynamic and energy aware authentication scheme for smart home appliances in Internet of Things (DAoT) is presented in [23]. DAoT focuses on authentication of identification of Internet of Things (loT) device for accessing loT network. Authors find key operations for authentication: key establishment (KE), message authentication code (MAC) operation and handshake. The KE operation securely derives confidential keys for cryptographic mechanisms. The MAC verifies integrity and authentication using the secret keys and cryptographic mechanisms [23]. However, anonymity and unlinkability are not the focus of the Kim et al scheme [23].

In [24], authors proposed a novel context-aware multi-attribute continuous authentication model for secure energy utilization management in smart homes. The scheme uses location and the critical nature of the tasks as the contextual information for supporting information allowing the selection of authentication attributes. The usefulness of the proposed solution is validated using real-world data sets.

A framework for maintaining security and preserving privacy for analysis of sensor data from smart homes is proposed in [25]. The main focus of Chakravorty et al is the data security instead of the device anonymity [25]. In [26], Ryu-Kwak proposed a secure data access control scheme for smart homes. The scheme [26] authenticates all devices registered to a smart home and provides safe access control of the data while excluded the unlinkability. Moreover, the traditional authentication protocols [27] [28] [29] [30] use two/three factor based authentication. However, the main focuses of the traditional authentication schemes are the human interventions by means of the password and/or biometric utilizations. The difference between these [27] [28] [29] [30] protocols and smart home setting is that the authentication procedure needs to be automatically activated by the devices (for instance, the appliances, sensors, actuators, etc.), which does not involve any human interaction.

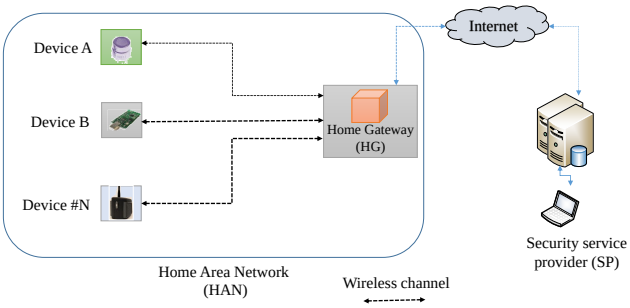

Fig. 1. System model for Home area network (HAN).

III. System Model, Attack Model and Assumptions

In this section, we formalize the system model for the smart home, notations and the assumptions used in the ASF.

1) System model: A smart home is a tiny intelligent world that provides services to the inhabitants. Our system model is similar to the scheme proposed in [18] [20] [31]. As shown in Fig. 1, we consider a typical home area network (HAN), which comprises of a number of heterogeneous devices (e.g., device A, device B, and so on) connected to a common gateway.

Home area network (HAN): This involves mainly three entities, the device (A), the home gateway (HG), and the service provider (SP), as follows.

- The device A (smart object and/or smart appliance) is integrated with the sensor network functionality. The device A is often restricted in terms of computational power, bandwidth, and memory, requiring very efficient operations from the side of the device.
- The home gateway (HG) is connected with a large number of smart devices, appliances, and with the outer world via the Internet. The HG, mainly performs two functions: aggregation and relaying. The aggregation component is responsible for collecting sensor data and controlling home devices, while the relaying component helps to transmit the device data to the individuals, when they are out of the home. For ease in notation, we here consider the situation that there is only one HG responsible for all the devices. Nevertheless, the protocol is described in such a way that extension is easily possible in the HAN use-cases since the identity of the HG is involved.
- As in [18], the security service provider (SP) is a trusted server, and is responsible for generating, distributing the secret keying material and cyrptosystems to the smart home devices and the HG.

Communication model: In the HAN, typically, the smart device communicates to the HG through the HAN protocol, e.g., ZigBee. Whereas, the HG utilizes mainly two wireless interfaces: (i) a short-range wireless interface (e.g., IEEE 802.15.4) maintains the connection with in the internal (smart) devices, and (ii) a long-range communication interface (e.g., Wi-Fi/GPRS) maintains a connection with the outer world via the Internet [18].

2) Attack model: We consider the Dolev-Yao attack model [32], where the attacker is able to eavesdrop on the traffic, inject new messages, replay and change messages, or spoof

TABLE I
SYMBOLS AND DESCRIPTIONS

Notations	Descriptions
SP	Security service provider
id_A	Identity of smart device A
id_G	Home gateway (HG) identity
$E_K[M]$	Message *m* is encrypted using secret key *(K)*
$D_K[M]$	Message *m* is decrypted using secret key *(K)*
α	Unique authentication token for the smart device A
$H()$	One way hash function
$\|\|$	Concatenation and XOR operation

other identities. In addition, the attacker may come from inside or outside the network. However, their goals might be to obtain illegitimate data access or control to the smart home devices, to perform service degradation or denial of service. It must be mentioned that a complete protection against these types of attacks is inherently very difficult. A minimal requirement is that detection mechanisms should be incorporated.

3) Assumptions:

- Consider a typical use-case in a smart home environment, where a climate sensor needs to provide its sensing information to the HG on a temporary basis or when substantial changes are notified.
- The SP and the HG are trusted entities and have no restrictions with respect to computation power and memory. In addition, both (the HG and SP) are considered to be tamper proof.
- The HG and the device A are having identical symmetric cryptographic systems, which are assumed to be secured (e.g., encryption, decryption and hash function).

Table I shows the notations and descriptions, which are used throughout the paper.

IV. ANONYMOUS SECURE FRAMEWORK (ASF)

This section proposes an anonymous secure framework for connected smart home. Different phases are distinguished: system setup, installation of devices, and the actual key establishment phase. We now discuss the construction of each of them into more detail, as follows.

A. System setup phase

This phase invokes offline. Let x and y be two high entropy secrets chosen by the SP. For a given HG with identity id_G, the SP computes $H(x\|y), H(id_G\|H(x))$. Finally, the SP stores secret parameters $y, id_G, H(x\|y), H(id_G\|H(x))$ to the memory of the HG.

B. Installation phase of the devices

Before deploying a smart device (e.g., Device A) into the HAN, it (Device A) should be registered and obtained secret credentials at the SP. In any other case, for a given device A with identity id_A, the SP computes

$$K = H(y\|H(id_G\|H(x))\|\alpha)$$
$$A_i = E_K(id_A\|N)$$
$$B_i = H(x\|y) \oplus A_i$$

Here, α is an unique authentication token and N denotes the number of times a device A with identity id_A requests an installation. If this number reaches a threshold, the SP may decide to refuse an installation. However, the basic idea behind this construction of the parameters A_i and B_i is that A_i should be only computable by the legitimate HG. The device A stores the parameters B_i and $H(A_i)$, which are used during the key agreement. Given the parameter B_i by the device A, the HG can derive A_i and thus also a shared value corresponding to $H(A_i)$. The reason why A_i is constructed by means of a key K, which compromises of information only known to one particular HG, is to avoid that this HG would be able to construct new devices for potentially other HGs in the field.

Finally, in order to conclude the installation phase, the device stores the values $id_G, H(x), H(A_i), B_i, \alpha, id_A$ in memory. The SP also can keep track of the identity of the device id_A, together with the parameters α and N.

C. Key establishment

The key establishment between the device A and the HG consists of three steps, containing of two communication passes and one final computation step. As assumed, the initiation of the protocol starts from the device A.

1) $A \rightarrow HG$: The device A generates a random number R_A and then it computes the following parameters.

$$V_1 = H(id_G\|H(x)) \oplus R_A \oplus T1$$
$$CID_i = B_i \oplus H(H(id_G\|H(x))\|R_A\|T1)$$
$$TK = H(A_i) \oplus R_A$$
$$C_1 = E_{TK}[id_A\|id_G\|N\|\alpha\|T1]$$

Here the device A derives a temporary key ($TK = H(A_i) \oplus R_A$). $T1$ is the current timestamp of the device A. Next, A sends $V_1, CID_i, C_1, T1$ to the HG.

2) $HG \rightarrow A$: Upon receiving the message, the HG starts following operations:

$$(T2 - T1) \leq \Delta T; \text{if not true then abort}$$
$$R_A = V_1 \oplus H(id_G\|H(x)) \oplus T1$$
$$B_i = CID_i \oplus H(H(id_G\|H(x))\|R_A\|T1)$$
$$A_i = B_i \oplus H(x\|y)$$
$$TK^* = H(A_i) \oplus R_A$$
$$D_{TK^*}[C_1] \text{ and obtain } id_A^*, id_G^*, N^*, \alpha^*, T1^*$$
$$\text{Check if } id_G == id_G^*, T_1 == T1^*$$
$$\text{Check } id_A^*, N^* \text{ by } A_i == E_K(id_A^*\|N^*)$$

If the checks on $id_A^*, id_G^*, N^*, T1^*$ are positive, the device A with real identity id_A is authenticated and included in the list of legitimate devices. Otherwise, the HG aborts the system. Here $T2$ is the current timestamp of the HG and ΔT is a transmission delay, which is agreed between the device A and HG. Next, the HG defines the shared symmetric key with the device A as $SK = H(R_A\|R_{HG}\|id_A\|id_G\|\alpha)$, with R_{HG} a random value chosen by the HG. In order to let the device A be able to derive this key, the following parameters are computed by the HG.

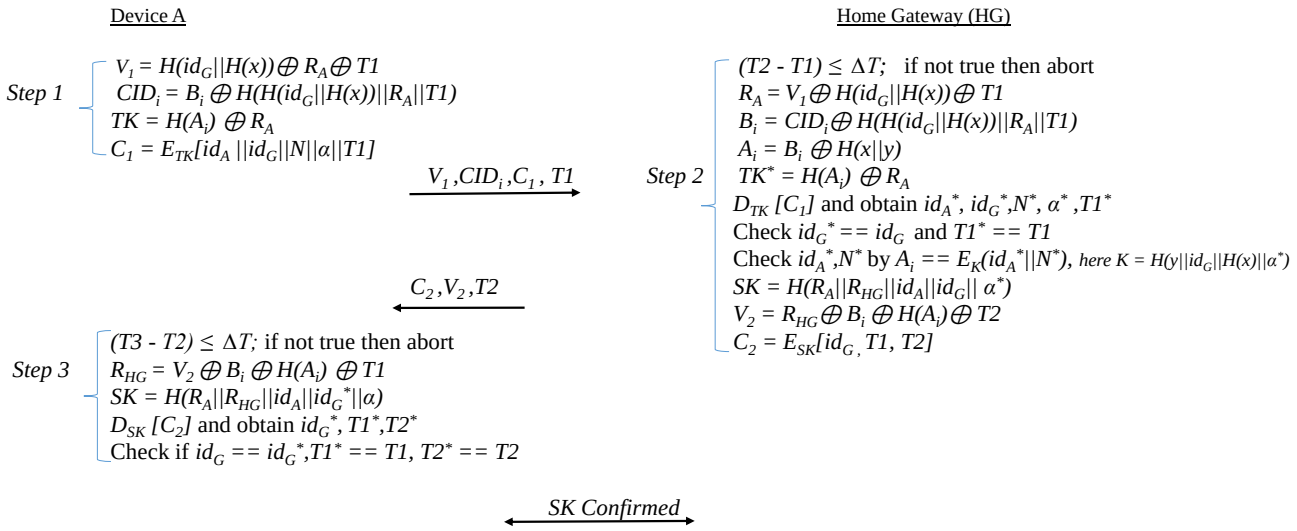

Fig. 2. Flow of the proposed ASF.

$$V_2 = R_{HG} \oplus B_i \oplus H(A_i) \oplus T2$$
$$C_2 = E_{SK}[id_G, T1, T2]$$

The HG sends $C_2, V_2, T2$ to the device A.

3) Key establishment: Upon receiving the message, the device A performs the followings.

$(T3 - T2) \le \Delta T$; if not true then abort
$R_{HG} = V_2 \oplus B_i \oplus H(A_i) \oplus T2$
$SK = H(R_A \| R_{HG} \| id_A \| id_G \| \alpha)$
$D_{SK}[C_2]$ and obtain $id_G^*, T1^*, T2^*$
Check if $id_G == id_G^*,\ T1 == T1^*,\ T2 == T2^*$

If three conditions are being verified, then the device A assures that the HG is an authentic gateway and the computed key $SK = H(R_A \| R_{HG} \| id_A \| id_G \| \alpha)$ can be used as the shared session key.

4) $A \leftrightarrow HG$: Finally, a *confirmed* message can eventually be sent by the device A to the HG.

After the key establishment, the device A shares a dynamic symmetric session key (SK) with the HG, which will be used to securely send its information to the HG. The flow of ASF is shown in Fig. 2.

With the help of above mentioned procedure, the new ASF can achieve the anonymity and unlinkability including authentication and integrity within the smart home network. The detailed analysis is discussed in Section VI.

V. Security Analysis of ASF

In this section: (a) we simulate the proposed ASF for formal security verification using the widely-accepted security analyzer tool, i.e., automated verification of Internet security protocol and application (AVISPA) tool [33] [18] [19]; (b) we formally analyse, e.g., authentication, session-key establishment and freshness of the proposed ASF using the well-known BAN-logic [34]; and (c) we informally analyse the security properties of the proposed ASF.

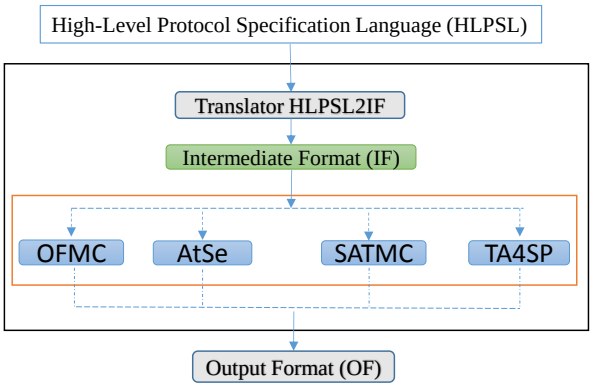

Fig. 3. Architecture for AVISPA tool [33] [19].

A. Simulation for formal security verification of ASF using AVISPA

AVISPA is a push-button security analyzer tool for the automated validation of Internet security-sensitive protocols and applications. The AVISPA tool consists of independently developed verification backends, as shown in Fig. 3. The backends are named as on-the fly model-checker (OFMC), constraint-logic-based attack searcher (CL-AtSe), SAT-based model-checker (SATMC), and tree automata based on automatic approximations of the analysis of security protocols (TA4SP). The tool uses a high level protocol specification language (HLPSL) for security protocol specification. As shown in Fig. 3, AVISPA tool takes HLPSL script as an input and translates to intermediate format (IF) using a HLPSL2IF translator. The translated IF code is the input for the backends, and finally the backends generate the output format (OF). HLPSL is an expressive, modular, formal language that allows for the specification of control flow patterns, data structures, alternative intruder models, complex security properties, as well as different cryptographic primitives and their algebraic properties. These features make HLPSL well suited for specifying modern, industrial-scale protocols.

Moreover, the HLPSL is a role-based language – it specifies the roles of each agent in a module that is called a *basic role*. The *basic role* represents what data the agent can use initially,

```
role deviceA (A, HG: agent,
              Hash : hash_func,
              K : symmetric_key,
              SK : symmetric_key,
              SND, RCV : channel(dy) )
played_by A def=
local State                                  :  nat,
       IDi, IDg, Ai, Bi, Alpha, T1, T2, X, Y  :  text,
        V1, V2, CIDi, TK, C1, C2              :  message,
       Hash                                   :  hash_func
const  deviceA_hg_Ra, hg_deviceA_Rg,
       deviceA_hg_IDi, hg_deviceA_IDg : protocol_id,
       deviceA_hg_T1, hg_deviceA_T2, sub1, sub2, sub3 : protocol_id
init State := 0
transition
1. State   = 0 /\ RCV(start) =|>
       State' := 1 /\ Ra' := new ()
                  /\  T1' := new ()
                 /\ V1' := xor(Hash(IDg.Hash(X)), xor(Ra',T1'))
                /\ CIDi' := xor(Bi, Hash(Hash(IDi.Hash(x)).Ra'.T1'))
                /\ TK' := xor(Hash(Ai),Ra')
                /\ C1 ' := {IDi.IDg.N.Alpha.T1'}_TK'
                /\ SND(V1',CIDi',C1',T1')
                /\ secret ({IDi, IDg}, sub1, {A, HG})
                /\ secret ({X, Y}, sub2, {A, HG})
                /\ secret (Alpha, sub2, {A, HG})
                /\ witness(A,HG,deviceA_hg_T1, T1')
                 % deviceA has freshly generated the value of T1 for HG
                /\ witness(A,HG,deviceA_hg_Ra, Ra')
                 % deviceA has freshly generated the value of Ra for HG

2. State = 3 /\ RCV ({IDg.T1.T2'}_SK, xor(Rg', xor(Bi,xor(Hash(Ai,T2')))), T2') =|>
```

Fig. 4. Specification of the deviceA role.

i.e., parameters, initial state, and ways in which the transition can happen. The *composition role* describes an entirely single session of the protocol by specifying how the legal agents are communicating to each other over the public channels. Moreover, a top-level role (i.e., *environment role*) contains global constants and a composition of one or more sessions, where the attacker may play some roles as a legitimate user. It also describes what knowledge the intruder has about the networks. In the AVISPA tool, the intruder is modeled through the *channel(dy)*. The *channel(dy)* uses the Dolev-Yao attack model, where an attacker can intercept, analyse, and modify the messages [32]. For more details refer to [33] [35].

1) Specifying ASF scheme: For the validation and testing, we have implemented the key establishment phase (refer Section IV.C) using HLPSL script. As we can see from Fig. 2 (i.e., key establishment phase), where the communication is taking place between the device A and the HG, therefore, there are two *basic roles*: deviceA and homeGateway, which are denoted by the deviceA and HG, respectively. For the deviceA, the role specifications are shown in Fig. 4 – the deviceA receives (RCV) a start signal then changes its initial state (i.e., 0) to 1 and sends *<V1, CIDi, C1, T1>* using *SND()* operation to the HG. In State 3, the deviceA receive a message $<\{IDg.T1.T2'\}_SK, xor(Rg', xor(Bi, xor(Hash(Ai, T2'))))$, $T2') >$ using *RCV ()* operation from the HG. Moreover, in the declaration part, *channel(dy)* denotes that the channel is for the attacker – the attacker/intruder can intercept, analyze, and modify the messages via channel eavesdropping.

Similarly, the role specifications of the homeGateway are shown in Fig. 5. The HG receives a message *<V1, CIDi, C1, T1>* from the deviceA then changes its initial state (i.e., 0) to 1 and sends $<\{IDg.T1.T2'\}_SK, xor(Rg', xor$ $(Bi, xor(Hash(Ai, T2'))), T2') >$ using *SND ()* operation to the devieA. Fig. 6 and Fig. 7, depict the roles of the session, and the environment and goals, respectively, of the proposed ASF. Fig. 6 shows the basic roles of session where the deviceA and homeGateway are instanced with concrete arguments, e.g., $deviceA(A, HG, SK, Hash, AS, AR)$ and $homeGateway(A, HG, SK, Hash, HGS, HGR)$. In Fig. 7, the top-level role (environment) is always defined that contains

```
role homeGateway (A, HG: agent,
              Hash : hash_func,
              K: symmetric_key,
              SK : symmetric_key,
              SND, RCV : channel(dy) )
played_by HG def=
local State : nat,
       IDi, IDg, Ai, Bi, Alpha : text,
       T1, T2, X, Y            : text,
       V1, V2, CIDi, TK, C1, C2 : message
const  deviceA_hg_Ra, hg_deviceA_Rg,
       deviceA_hg_IDi, hg_deviceA_IDg : protocol_id,
       deviceA_hg_T1, hg_deviceA_T2, sub1, sub2, sub3 : protocol_id
init State := 0
transition
1. State = 0 /\ RCV(V1',CIDi',C1',T1') =|>
    State' :=  1       /\  T2' := new ()
               /\  secret ({Hash(X, Y)}, sub2, {A, HG})
               /\  secret ({IDi, IDg}, sub1, {A, HG})
               /\  secret ({Hash(X. Y)}, sub2, {A, HG})
               /\  Rg' := new ()
               /\ SK' := Hash(Ra'.Rg',IDa.IDg.Alpha)
               /\  xor(Rg', xor(Bi,xor(Hash(Ai,T2'))))
               %/\  ({IDg, T1, T2'}_SK')
               /\  SND({IDg.T1.T2'}_SK, xor(Rg', xor(Bi,xor(Hash(Ai,T2')))), T2')
               /\ witness ( HG, A, hg_deviceA_T2, T2')
                % HG has freshly generated the value of T2 for deviceA
```

Fig. 5. Specification of the homeGateway role.

```
role session (A, HG: agent,
              Hash : hash_func,
             K: symmetric_key ,
             SK : symmetric_key)
def=
local AS, AR, HGS, HGR: channel (dy)
composition
        deviceA(A,HG, SK, Hash, AS, AR)
        /\ homeGateway(A, HG, SK, Hash, HGS,HGR)
```

Fig. 6. Specification of the Session role.

the global constants and the composition of sessions (e.g., one or more sessions). In the sessions, the attacker/intruder may play some roles as legitimate users. In our specification, the intruder also participates in the execution of protocol as a concrete session (i.e., intruder_knowledge = {deviceA, homeGateway, h}), refer Fig. 7. Moreover, as shown in Fig. 7, the two secrecy goals and four authentications are verified in the proposed ASF, as follows:

- Secrecy_of sub1, represents that $\{IDi, IDg\}$ are kept secret between the deviceA and HG.
- Secrecy_of sub2, represents that $\{Hash(x\|y)\}$ are kept secret between the deviceA and HG.
- Authentication_on deviceA_HG_IDi, states that the device A identity (i.e, IDi) will be verified at the HG.
- Authentication_on HG_deviceA_IDg, states that the HG identity (i.e, IDg) will be verified at the device A.
- Authentication_on deviceA_HG_T1, states that the de-

environment.hlpsl

```
role environment()
def=
const deviceA, homeGateway: agent,
       sk : symmetric_key,
       h : hash_func,
       xg, yg, k, alpha, ida, idg, ra, rg, t1, t2 : text,
       deviceA_hg_Ra, hg_deviceA_Rg, deviceA_hg_IDi, hg_deviceA_IDg
       deviceA_hg_T1, hg_deviceA_T2, sub1, sub2, sub3 : protocol_id
intruder_knowledge = {deviceA, homeGateway, h}
composition
session(deviceA, homeGateway, h)
/\ session(deviceA, i, h)
/\ session(homeGateway, i, h)
end role

goal
secrecy_of sub1
secrecy_of sub2
%secrecy_of sub3
authentication_on deviceA_hg_IDi
authentication_on hg_deviceA_IDg
authentication_on deviceA_hg_T1
```

Fig. 7. Specification of the goal and environment for the proposed ASF.

```
Output
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/ASF.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.01s
  visitedNodes: 16 nodes
  depth: 4 plies
```

Fig. 8. ASF results using OFMC backend.

vice A timestamp (i.e, T1) will be verified at the HG.
- Authentication_on HG_deviceA_T2, states that the HG timestamp (i.e, T2) will be verified at the device A.

Fig. 8 shows the formal simulation and verification results of the proposed ASF using OFMC backend. The simulation results ensure that the proposed scheme is safeguard against the active attacks, for instance, the replay and man-in-the-middle attacks, and passive attacks. The summary of the results under OFMC reports that the ASF is safe.

B. ASF formal proof-using BAN logic

In this sub-section, we present the formal analysis (e.g., authentication, session-key establishment and freshness) of the proposed ASF using the well-known BAN-logic [34]. For details, the reader may refer to [34].

BAN logic notations and rules: We use directly BAN-logic symbols and notations from [34] to verify the proposed framework, as follows.

- $X \mid\equiv Y$: Let X and Y two principal entities and in BAN-logic '*X believes Y*'.
- $X \lhd Y$: Only '*X sees Y*', i.e., assume someone has sent a message containing Y to X, then X can read and repeat Y (i.e., after performing some decryption).
- $X \mid\sim Y$: The principal '*X once said Y*', i.e., at some time the principal X sent a message including Y.
- $X \mid\Rightarrow Y$: The principal '*X has control over Y*', the principal X is an authority on Y and should be trusted (For instance a server is often assumed to be trusted and to distribute secret keys efficiently and properly).
- $X \overset{M}{\Leftrightarrow} Y$: The principal X and Y have message (M) that contains the secret parameters.
- $\sharp(Y)$: *Fresh(Y)*, which means that Y has not been sent recently in a message during the protocol execution and *Fresh(Y)* protects from replay attack.
- $X \overset{\mathrm{K}}{\longleftrightarrow} Y$: The X and Y used a secret key K for securing the communication. It is assumed that key K will never be disclosed, except to the designated legitimate principals.
- $\{M\}_K$: Message M is encrypted using the secret key K.
- $\langle M \rangle_N$: i.e., M is combined with the secret parameter N and or with the identity.

Logical rules The following logical rules will be used to verify the protocol [34].

1) *Message-meaning rule*

$$\frac{X \mid\equiv Y \overset{\mathrm{K}}{\longleftrightarrow} X,\ X \lhd \{M\}_K}{X \mid\equiv Y \mid\sim M}$$

2) *Nonce-verification rule*

$$\frac{X \mid\equiv \sharp(M),\ X \mid\equiv Y \mid\sim M}{X \mid\equiv Y \mid\equiv M}$$

3) *Control rule*

$$\frac{X \mid\equiv Y \mid\Rightarrow M,\ X \mid\equiv Y \mid\equiv M}{X \mid\equiv M}$$

4) If a principal sees a formula, then it also sees its components, provided he knows the necessary keys

$$\frac{X \lhd \langle M \rangle_N}{X \lhd M},\ \frac{X \lhd (M, N)}{X \lhd M}$$

5) *Fresh rule*

$$\frac{X \mid\equiv \sharp(M)}{X \mid\equiv \sharp(M, N)}$$

If one part of a formula is fresh, then the entire formula must also be fresh [34].

Formal verification of the proposed ASF: We present the formal verification of the proposed ASF. Using the BAN logic, our formal analysis mainly focuses on the belief and freshness, consisting of the following steps: (i) message formalization, (ii) initial assumptions, (iii) expected goals, and (iv) logic verification.

(i) ***Message idealization***: Message idealization is to specify the exchanged messages. In the ASF, the idealized messages among the device A and the HG are obtained as follows.

$$\begin{aligned}
M1:\ & HG \lhd V_1\ (= H(id_G \| H(x)) \oplus R_A \oplus T1)\\
& HG \lhd CID_i\ (= B_i \oplus H(H(id_G \| H(x)) \| R_A \| T1)\\
& HG \lhd C_1\ \{id_A \| id_G \| N \| \alpha \| T1\}_{TK}\\
& HG \lhd T1\\
M2:\ & A \lhd V_2\ (= R_{HG} \oplus B_i \oplus H(A_i) \oplus T2)\\
& A \lhd C_2\ (= \{id_G, T1, T2\}_{SK})\\
& A \lhd T2
\end{aligned}$$

(ii) ***Assumptions***: In ASF, a principal believes that (a) it has shared secrets and keys with the assigned principals, (b) the random numbers and timestamps are fresh, and (c) a legitimate principal has control over the entitled components and values. The intuitive assumptions are as follows:

- For the HG:

A1: $HG \mid\equiv A \overset{V_1, CID_i, C_1, T1}{\longleftrightarrow} HG$
A2: $HG \mid\equiv A \overset{SK}{\longleftrightarrow} HG$
A3: $HG \mid\equiv (A \mid\Rightarrow A \overset{SK}{\longleftrightarrow} HG)$
A4: $HG \mid\equiv \sharp(R_{HG})$
A5: $HG \mid\equiv \sharp(T2)$

A6: $HG \mid\equiv (A \mid\Rightarrow id_A)$

- For the device A:

A7: $A \mid\equiv HG \overset{C_2,V_2,T2}{\longleftrightarrow} A$
A8: $A \mid\equiv HG \overset{K}{\longleftrightarrow} A$
A9: $A \mid\equiv HG \overset{SK}{\longleftrightarrow} A$
A10: $A \mid\equiv (HG \Rightarrow HG \overset{SK}{\longleftrightarrow} A)$
A11: $A \mid\equiv \sharp\ (R_A)$
A12: $A \mid\equiv \sharp\ (T1)$

(iii) ***Expected goals***: The expected security goals refer to belief and freshness of the exchanged messages, which are transmitted between the legitimate entities and agreed on a session key (SK). Moreover, the messages are never used in former sessions. In ASF, the expected goals are the following.

G1: $HG \mid\equiv A \overset{K}{\longleftrightarrow} HG$ i.e., shared secret key (K)
G2: $A \mid\equiv HG \mid\equiv A \overset{SK}{\longleftrightarrow} HG$ i.e., session key (SK)
G3: $HG \mid\equiv A \mid\equiv HG \overset{SK}{\longleftrightarrow} A$ i.e., session key (SK)
G4: $HG \mid\equiv \sharp\ (R_A, T1)$ i.e., freshness
G5: $A \mid\equiv \sharp\ (R_{HG}, T2)$ i.e., freshness

(iv) ***Logic verification***: Based on the message idealization, and initial assumptions, and BAN-logic rules, we perform the logic verification (i.e., expected goals).

Goal 1: $HG \mid\equiv A \overset{K}{\longleftrightarrow} HG$

Proof: According to A1: $HG \mid\equiv A \overset{C_1,V_1,CID_i,T1}{\longleftrightarrow} HG$, it turns out that the HG believes the device A, which wants to start a session.

Now, according to M1: $HG \lhd V_1,\ HG \lhd CID_i,\ HG \lhd C_1,\ HG \lhd T1$, it turns out that the HG *sees* $\langle V_1 \rangle_{\sharp RA}$, $\langle CID_i \rangle_{Bi}$, therefore, $HG \lhd TK\ (= H(A_i) \oplus R_A)$, and $\{C_1\}_{TK}$, by applying rule 4. Due to $id_G == id_G^*$, $T_1 == T1^*$, we obtain that

$$HG \ \lhd\ \{C_1\}_{TK}, \langle CID_i \rangle_{Bi},\ \langle V_1, CID_i \rangle_{\sharp RA}$$

Next, applying *message-meaning rule* and A6, and the following is obtained

$$HG \mid\equiv A \mid\sim \langle CID_i \rangle_{B_i}$$
$$HG \mid\equiv A \mid\sim \langle C_1, CID_i \rangle_{idA, \sharp RA}$$
$$HG \mid\equiv A \mid\sim \langle V_1 \rangle_{\sharp RA}$$

Note that, B_i contains $H(x\|y) \oplus A_i$ and here $A_i\ (= E_K(id_A\|N))$ is computed using a shared key K, refer installation phase of the devices (Section IV.B). It turns out that

$$HG \lhd A_i\ (= \{id_A\|N\}_K)$$

Therefore,

$$HG \ \mid\equiv A \overset{K}{\longleftrightarrow} HG$$

If HG believes that key K is a shared secret with device A and then the HG will believe that the device A once sent the message (i.e., $C_1, V_1, CID_i, T1$). Moreover, if the HG believes that $id_A == id_A^*$ and $N == N^*$ then the device A is the legitimate and authenticated entity. □

Goal 2: $A \mid\equiv HG \mid\equiv A \overset{SK}{\longleftrightarrow} HG$

Proof: According to A7: $A \mid\equiv HG \overset{C_2,V_2,T2}{\longleftrightarrow} A$, it turns out that the device A believes the HG and the message $C_2, V_2, T2$ contains the secret parameters, which will be used to derive the session key (SK).

Considering M2: $A \lhd C_2$, $A \lhd V_2$ and $A \lhd T2$, it turns out that the device A receives $(R_{HG} \oplus B_i \oplus H(A_i) \oplus T2)$, $\{id_G, T1, T2\}_{SK}$, and $T2$. Due to $id_G^* = id_G$, we obtain that

$$A \ \lhd\ \langle V_2 \rangle_{\sharp RHG}$$
$$A \ \lhd\ \sharp\ (T2)$$

By applying message-meaning rule and A12, we obtain

$$A \mid\equiv HG \mid\sim \langle V_2 \rangle_{\sharp R_{HG}}$$
$$A \mid\equiv HG \mid\sim \langle SK \rangle_{\sharp RA,\ \sharp RHG,\ idA,\ id_G, \alpha}$$
$$A \mid\equiv HG \mid\sim \sharp\ (T2)$$

If the device A believes the HG and so the $\sharp R_{HG}$ and $\sharp T2$ then it also believes in the session key (SK). By applying A9 and A10, we obtain

$$A \mid\equiv HG \mid\equiv A \overset{SK(=H(R_A\|R_{HG}\|id_A\|id_G\|\alpha))}{\longleftrightarrow} HG$$
$$A \mid\equiv (HG \Rightarrow HG \overset{SK}{\longleftrightarrow} A)$$

Thus, G2 has proven, and similarly, G3 can be proved. □

Goal 4: $HG \mid\equiv \sharp\ (R_A, T1)$

Proof: From M1: $HG \ \lhd\ V_1$, in which V_1 contains $H(id_G\|H(x)) \oplus R_A \oplus T1$, here R_A is a random number and HG will compute $R_A = V_1 \oplus H(id_G\|H(x)) \oplus T1$. Now, applying rule 5 (fresh rule) and A11, we obtain that,

$$HG \mid\equiv \sharp\ (R_A)$$

Next applying to M1: $HG \lhd T1$, here $T1$ is the current timestamp of device A. Applying rule 5 and A12, we obtain,

$$HG \mid\equiv \sharp\ (T1)$$

Hence the goal G4 ($HG \mid\equiv \sharp\ (R_A, T1)$) has been achieved, i.e., the HG believes that $\sharp R_A$ and $\sharp T1$ are fresh. Similarly, the goal G5 can be proved, i.e., $A \mid\equiv \sharp(R_{HG}, T2)$. □

C. Informal (security) analysis:

This sub-section discusses the security properties of the ASF along with the prevention against possible security attacks. Recall the attack model from Section III.B, where an adversary is able to eavesdrop on the traffic, inject new messages, replay and change messages, or spoof identities.

Anonymity and unlinkability: Assume that an adversary (J) eavesdrops on the wireless traffic between the device A and the HG and tries to spoof the device's identity id_A. However, in the proposed ASF, when the device A wants to connect to the HG, it does not send id_A in the plain-text but a temporary $CID_i\ (= B_i \oplus H(H(id_G\|H(x))\|R_A\|T1))$ *alias* instead. Only the legitimate HG can deduce the real identity of the device A by decrypting $C_1 = E_{TK}[id_A\|id_G\|N\|T1]$, $A_i = E_K(id_A\|N)$. The proposed framework therefore achieves identity anonymity between the device A and the HG.

Similarly, the proposed ASF achieves unlinkability. Assuming that the adversary tries to trace whether a legal device A has previously requested to connect the HG, the attacker will not be able to link this attempt successfully in the ASF. For instance, for each session, the device A computes the authentic message $(C_1, V_1, CID_i, T1)$ in step 1, cf. Fig. 2. All these messages are always different, since the message $(C_1, V_1, CID_i, T1)$ is randomized by the random number R_A and/or it contains the current timestamp $(T1)$ of the device A. It is difficult to link two different value instances $(C_1, V_1, CID_i, T1)$. Likewise, the same holds for step 2, cf. Fig. 2. The ASF therefore achieves unlinkability.

However, it should be noted that unlinkability does hold for the malicious devices connected to the same HG. Assuming that a malicious device may retrieve (somehow) B_i from CID_i as it is aware of $H(id_G \| H(x))$. However, from $B_i (= H(x \| y) \oplus A_i)$, it is still not possible to derive the corresponding device A's identity because (A_i) is encrypted with key $K (= H(y \| H(id_G \| H(x)) \| \alpha))$, refer Section IV-B.

Mutual authentication and integrity: Mutual authentication is an important property for a verification service resisting to unauthorized access. The proposed ASF provides a mutual authentication for the communicating entities. The HG can authenticate the device A by means of id_A^*, N^* from $A_i == E_K(id_A^* \| N^*)$, refer step 2 in Fig. 2. Similarly, the device A can authenticate the HG using $id_G == id_G^*$.

In addition, message integrity is realized by one-way hash functions. The device A messages (CID_i) are computed and transmitted in terms of $H(\cdot)$ for identifying declaration and verification of the ASF messages.

Resistance to replay attack: In these attacks, an adversary (J) wants to replay the previous messages, which are being eavesdropped from the communication entities, e.g., the device A and the HG. Assuming that an attacker intercepts valid message $(C_1, V_1, CID_i, T1)$ and tries to start a session with the HG by replaying the same intercepted message. The message verification at the HG will fail due to the interval $(T2 - T1) \leq \Delta T$, here $T2$ is HG's system time while receiving the replayed message. Similarly, the same holds when an attacker (J) intercepts a valid message $(C_2, V_2, T2)$ and tries to connect to the device A by replaying the same intercepted message. The message verification at the device A will fail because of the interval $(T3 - T2) \leq \Delta T$, here $T3$ is the device A's system time while receiving the replayed message. Thus, the proposed ASF resistant to replay attacks.

Protection against impersonation attack: An attacker (J), willing to impersonate the device A, may try to forge the message $(C_1, V_1, CID_i, T1)$ in step 1 in Fig. 2. However, J can not know the device's real identity, i.e., id_A, which is confidential $(E_{TK}[id_A \| id_G \| N \| T1]$ and $E_K(id_A \| N))$ under the temporary key $(TK = H(A_i) \oplus R_A)$ and secret key K $(= H(y \| H(id_G \| H(x)) \| \alpha))$, respectively. Since the key $K (= H(y \| H(id_G \| H(x)) \| \alpha))$ is computed using the unique authentication token (α), the attacker cannot impersonate the device A by forging a correct message.

Similarly, to impersonate the HG, an attacker must know the secret token (α) to generate the legal message (C_2), which is encrypted with $SK = H(R_A \| R_{HG} \| id_A \| id_G \| \alpha)$. Since, J does not possess secrets (α), he/she cannot impersonate the legitimate HG.

Resistance against man-in-the-middle (MITM) attack: We here assume that an attacker (J) may try a MIMT attack by modifying $(C_1, V_1, CID_i, T1)$ to $(C_{1J}, V_{1J}, CID_{iJ}, T1)$. The HG however will detect this attempt when the parameters of A_i are decrypted using $K = H(y \| H(id_G \| H(x)) \| \alpha)$ and will verify id_A. If this does not hold, the HG aborts the system. Moreover, as J does not know the key K, he/she therefore cannot compute the real $A_i = E_K(id_A \| N))$. Thus MITM attack is difficult to the proposed ASF.

Secure session key agreement with forward secrecy: It can be seen in the proposed ASF, after performing the mutual authentication between the device A and the HG, the session key $(SK = H(R_A \| R_{HG} \| id_A \| id_G \| \alpha))$ is being generated using pseudo-random numbers and timestamps to provide session freshness and randomization. The transmitted messages are typically computed using the random numbers (R_A, and R_{HG}), which make that the exchanged messages can be regarded as dynamic variables. Moreover, compromising long-term key K does not compromise past sessions because the adversary has no way to obtain the random numbers (R_A, and R_{HG}), which are protected in $V_1 = H(id_G \| H(x)) \oplus R_A \oplus T1$, $CID_i = B_i \oplus H(H(id_G \| H(x)) \| R_A \| T1)$, and $V_2 = R_{HG} \oplus B_i \oplus H(A_i) \oplus T2$. Therefore, the ASF provides perfect forward secrecy and an adversary will find difficulties to correlate the ongoing session with the previous sessions.

Secure against smart device compromised threat: Assume that an attacker compromises the smart device (e.g., device A) and tries to get secret information from the device. It is widely accepted that physical attacks are difficult to prevent if the devices are not tamper-proof [36]. Nevertheless, similar to [18], the proposed ASF relies on the deviceA-to-HG communication architecture, where each smart device stores a unique authentication token (α), which is shared with the HG only. Therefore, no communication is being taking place between two distinct smart devices [18], which means the proposed scheme can increase the network resilience against a node (e.g., device A) compromised threat. Furthermore, in the smart home settings, the smart devices are physically secured since they are deployed mostly inside the home where the HG can check at regular interval whether a smart device is misbehaving using the scheme proposed in [37].

VI. Performance analysis

In this section, we analyze the performance of the proposed ASF in terms of computational and communicational costs.

Experiment setup: In order to implement the ASF on the low-powered device (i.e., device A), we choose TelosB mote, same as in [18]. TelosB mote [38] runs TinyOS version 2.x [39]. We recommend the AES (Advanced Encryption Standard) symmetric-key algorithm for the encryption. AES is the current encryption standard and it is integrated in CC2420 radios [40]. For the sake of experiment purposes, we chosen SHA1 [41] for the hashing operations. Similar to [18], the HG is considered to be the clock manager for the smart home and synchronizes the clocks of the smart devices. To

In bytes	2	1	4	variable	variable	3	4
	Frame control	Seq. no.	Add. fields	Super frame	Beacon payload	Clock frequency	Synchronizing information

Fig. 9. Beacon frame format [18].

TABLE II
MEMORY CONSUMPTION AND EXECUTION TIME

Operations	RAM *in KB*	ROM *in KB*	Time
Hash	1.3	10	39 *(in* ms*)*
Encryption	1.9	9.4	3.5 *(in* ms*)*
Decryption	–	–	41.15 *(in* ms*)*
XOR	–	–	106 *(in* ps*)*

RAM- Random access memory, and ROM - Read only memory.

do this, the HG sends a time beacon frame (which includes clock frequency and synchronization information, as shown in Fig. 9 [18]) to the home devices. Interested readers may follow [18] for the detailed experimental setup. Moreover in our experimental setup, we consider a typical use-case for the smart home environment, where a sensor needs to provide its sensing information to the HG on a temporary basis.

Computational cost: Before starting the investigation, note that we consider the communication cost only for the device A, since it is a resource-hungry device. To analyze the computational price (in terms of memory consumption and execution time), we consider only the key establishment phase (refer Section IV-C). As we can see from Fig. 2, the proposed framework is mainly based on the hash function, encryption, decryption, and XOR operation. Table II shows that the ASF requires reasonable memory size and execution time for the TelosB mote. Bitwise Exclusive-OR operations use only bits shifting, and thus have negligible memory footprints. Furthermore, Table III shows computational cost comparisons of the proposed ASF with [17], [18] and [20] schemes. It can be seen from Table III, the proposed ASF requires 2H + 1E +1D + 3XOR operations, which are lightweight in terms of execution time for such resource-hungry smart home devices. Whereas, the public key cryptography operations, like the signature generation and the point multiplication are quite expensive in terms of time complexity [42]. The schemes proposed in [17] and [20] take 2t + 4H + 1Sig and 2t + 1H + 1Sig, respectively, therefore, both schemes are expensive for the low-cost smart devices (e.g., TelosB). Kumar et al scheme [18] requires 2H + 1MAC + 1HMAC + 1E + 1D operations, which is also well suited to the home environments. In addition, Table IV briefly compares the time complexity of our framework with the existing schemes. The actual time complexity depends upon the time taken to execute each operation at the low-powered device (e.g., TelosB) by each protocol. Considering [42] implementations, Vaidya et al [17] scheme incurs ≈ 10.336 $seconds$, and Santoso-Vun's scheme requires ≈ 10 $seconds$. Kumar et al's scheme incurs ≈ 0.17 $seconds$ to execute the whole protocol at the device A. Whereas the ASF requires ≈ 0.123 $seconds$. Due to the fact of public key cryptography, [17] and [20] are quite expensive, while on the contrary [18] and the proposed ASF requires significantly low computation cost.

TABLE III
COMPUTATION COST COMPARISONS AT RESOURCE-HUNGRY DEVICE

	[17]	[18]	[20]	ASF
Point multiplication	2t	–	2t	–
Hash operation	4H	2H	1H	2H
MAC	–	1MAC	–	–
HMAC	–	1HMAC	–	–
XOR operation	–	–	–	3XOR
Cryptosystem	–	1E+1D	–	1E+1D
Signature	1 Sig	–	1 Sig	–

t - the time for executing point-multiplication; H - the time for executing one-way hash function; E - the time for performing encryption; D - the time for performing decryption; MAC - the time for performing MAC operation; HMAC - the time for performing HMAC operation; XOR - the time for performing XOR operation, and Sig - the time for generating signature.

TABLE IV
TOTAL EXECUTION TIME (IN SECONDS)

	[17]	[20]	[18]	ASF
Total executing time	$\approx$ 10.336	$\approx$ 10	$\approx$ 0.17	$\approx$ 0.123

However, it is important to note that all the previous protocols have different security services from the proposed ASF. For instance, the protocols proposed in [17] [18] [20] are mainly providing authentication and session key establishment between the communicating entities, whereas the new ASF is also providing anonymity and unlinkability including authentication and key establishment properties to preserve the smart home devices security and privacy.

Communicational cost: To investigate the communicational overhead of the ASF, we consider the total number of bits transmitted and received by the device A to start the bootstrapping. Total communication cost of the proposed ASF is as follows. In [18], authors suggested the following message sizes for the smart home environment, device A ID as 1 byte, random number as 4 bytes, timestamp as 4 bytes, and 16 bytes key size, and 16 bytes for a hash function. Consider similar packet sizes, the first message sends 232 bits to the HG (i.e., $A \rightarrow HG$) in ASF, and receives 72 bits of message (i.e., $HG \rightarrow A$) from the HG. Resultant, the communication overhead for the proposed framework is 304 bits (i.e., 232 + 72). Considering similar home environment, a smart home device transmits one message (256 bits), and receive two messages (268 bits) in Kumar et al's scheme [18]. We did not consider the communication costs (in bits) for the Vaidya et al's and Santoso-Vun's schemes, since the authors did not implement their schemes. Table V summarizes the communication overhead for the proposed ASF and [18].

Moreover, for the sake of comparison purposes, Fig. 10 shows the communication costs comparisons (in terms of the number of message exchanges) of the proposed scheme and [17], [20], [18]. As we can see from Fig. 10, to execute the whole protocol, Vadiya et al scheme [17] takes two rounds of message exchanges, Santoso-Vun's and Kumar et al's schemes need three message exchanges. Whereas, the proposed ASF requires two rounds of message exchanges. It indicates that ASF's computation and communication costs are reasonable.

TABLE V
COMMUNICATION OVERHEAD

	[18]	Proposed ASF
Send (in bits)	1S = 296	232
Receive (in bits)	2R = 268 (i.e., 108 + 160)	72
Total (in bits)	564	304

S - messages send; R - messages receive.

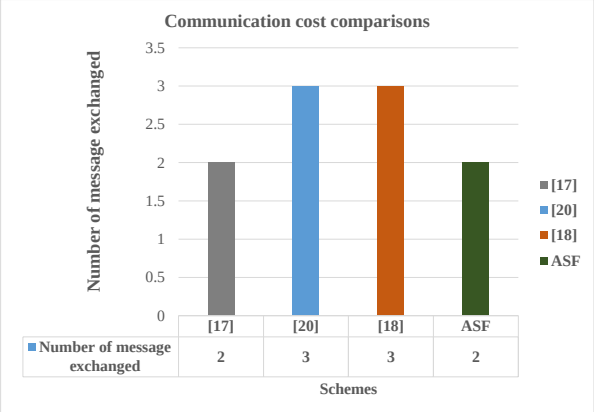

Fig. 10. Communication cost comparisons.

VII. CONCLUSION

Connected smart home environments offer enriched services and information for individuals. Such homes are heterogeneous and dynamic: they contain smart devices to enable individuals to enjoy network based services, such as climate control, energy management, home healthcare, and so on. However, device anonymity and unlinkability are actual challenges, where an unauthorized entity can identify the home devices (e.g., appliances, etc.) identities, sensors presence, and data-collection activities by network tracking.

This paper therefore defined a set of desirable properties for securing the smart home environments and presented an anonymous secure framework (ASF) for the connected smart homes. The proposed framework realized anonymity and unlinkability, authentication and integrity, established mutual trust relationships via the lightweight operations, and achieved session freshness dynamically. It indicated that the ASF is suitable for the next-generation smart home environments.

ACKNOWLEDGMENT

This work has received funding from the Research Council of Norway (PREAPP project, grant no. 231746/F20); the part of work acknowledged the contribution of the COST Action IC1303 - AAPELE, Architectures, Algorithms and Platforms for Enhanced Living Environments"; and the part of this work is also supported by Academy of Finland project SECUREConnect (296693).

REFERENCES

[1] Y. T. Lee, W. H. Hsiao, C. M. Huang, and S. C. T. Chou, "An integrated cloud-based smart home management system with community hierarchy," *IEEE Transactions on Consumer Electronics*, vol. 62, no. 1, pp. 1–9, February 2016.

[2] J. E. Kim, G. Boulos, J. Yackovich, T. Barth, C. Beckel, and D. Mosse, "Seamless integration of heterogeneous devices and access control in smart homes," in *Proceedings of the 2012 Eighth International Conference on Intelligent Environments*, ser. IE '12, 2012, pp. 206–213.

[3] C. Gomez and J. Paradells, "Wireless home automation networks: A survey of architectures and technologies," *IEEE Communications Magazine*, vol. 48, no. 6, pp. 92–101, 2010.

[4] A. Kailas, V. Cecchi, and A. Mukherjee, "A survey of communications and networking technologies for energy management in buildings and home automation," *Journal of Computer Networks and Communications*, vol. 2012, 2012.

[5] K. Gill, S.-H. Yang, F. Yao, and X. Lu, "A zigbee-based home automation system," *IEEE Transactions on Consumer Electronics*, vol. 55, no. 2, pp. 422–430, 2009.

[6] J. Kim, E. S. Jung, Y. T. Lee, and W. Ryu, "Home appliance control framework based on smart TV set-top box," *IEEE Transactions on Consumer Electronics*, vol. 61, no. 3, pp. 279–285, Aug 2015.

[7] B. C. Choi, S. H. Lee, J. C. Na, and J. H. Lee, "Secure firmware validation and update for consumer devices in home networking," *IEEE Transactions on Consumer Electronics*, vol. 62, no. 1, pp. 39–44, February 2016.

[8] "The Smart Home: intelligent home automation," https://www.cleverism.com/smart-home-intelligent-home-automation/.

[9] N. E. Petroulakis, I. G. Askoxylakis, and T. Tryfonas, "Life-logging in smart environments: challenges and security threats," in *2012 IEEE International Conference on Communications (ICC)*. IEEE, 2012, pp. 5680–5684.

[10] G. W. Hart, "Nonintrusive appliance load monitoring," *Proceedings of the IEEE*, vol. 80, no. 12, pp. 1870–1891, Dec 1992.

[11] Z. Wang and G. Zheng, "Residential appliances identification and monitoring by a nonintrusive method," *IEEE transactions on Smart Grid*, vol. 3, no. 1, pp. 80–92, 2012.

[12] N. P. Hoang and D. Pishva, "A TOR-based anonymous communication approach to secure smart home appliances," in *2015 17th International Conference on Advanced Communication Technology (ICACT)*. IEEE, 2015, pp. 517–525.

[13] K. Han, J. Kim, T. Shon, and D. Ko, "A novel secure key paring protocol for rf4ce ubiquitous smart home systems," *Personal and ubiquitous computing*, vol. 17, no. 5, pp. 945–949, 2013.

[14] Y. Li, "Design of a key establishment protocol for smart home energy management system," in *Computational Intelligence, Communication Systems and Networks (CICSyN), 2013 Fifth International Conference on*. IEEE, 2013, pp. 88–93.

[15] M. Burrough and J. Gill, "Smart Thermostat Security: Turning up the Heat," http://www.burrough.org/Documents/Thermostat-final-paper.pdf.

[16] S. Guillet, B. Bouchard, and A. Bouzouane, "Correct by construction security approach to design fault tolerant smart homes for disabled people," *Procedia Computer Science*, vol. 21, no. 0, pp. 257 – 264, 2013.

[17] B. Vaidya, D. Makrakis, and H. Mouftah, "Device authentication mechanism for smart energy home area networks," in *Consumer Electronics (ICCE), 2011 IEEE International Conference on*, Jan 2011, pp. 787–788.

[18] P. Kumar, A. Gurtov, J. Iinatti, M. Ylianttila, and M. Sain, "Lightweight and secure session-key establishment scheme in smart home environments," *IEEE Sensors Journal*, vol. 16, no. 1, pp. 254–264, 2016.

[19] A. K. Das, "A secure and effective biometric-based user authentication scheme for wireless sensor networks using smart card and fuzzy extractor," *International Journal of Communication Systems*, 2015.

[20] F. K. Santoso and N. C. Vun, "Securing iot for smart home system," in *2015 International Symposium on Consumer Electronics (ISCE)*. IEEE, 2015, pp. 1–2.

[21] E. Ayday and S. Rajagopal, "Secure Device Authentication Mechanisms for the Smart Grid-Enabled Home Area Networks," Tech. Rep., 2013.

[22] J. Logue, S. Supramaniam, O. Hardison, and J. Luxenberg, "Multi-tiered authentication methods for facilitating communications amongst smart home devices and cloud-based servers," Jan. 12 2016, uS Patent 9,237,141. [Online]. Available: https://www.google.com/patents/US9237141

[23] Y.-P. Kim, S. Yoo, and C. Yoo, "DAoT: Dynamic and energy-aware authentication for smart home appliances in Internet of Things," in *2015 IEEE International Conference on Consumer Electronics (ICCE)*. IEEE, 2015, pp. 196–197.

[24] U. S. Premarathne, "Reliable context-aware multi-attribute continuous authentication framework for secure energy utilization management in smart homes," *Energy*, vol. 93, Part 1, pp. 1210 – 1221, 2015.

[25] A. Chakravorty, T. Wlodarczyk, and C. Rong, "Privacy preserving data analytics for smart homes," in *Security and Privacy Workshops (SPW), 2013 IEEE*. IEEE, 2013, pp. 23–27.
[26] H.-S. Ryu and J. Kwak, "Secure data access control scheme for smart home," in *Advances in Computer Science and Ubiquitous Computing*. Springer, 2015, pp. 483–488.
[27] S. Banerjee, M. P. Dutta, and C. Bhunia, "An improved smart card based anonymous multi-server remote user authentication scheme," *International Journal of Smart Home*, vol. 9, no. 5, pp. 11–22, 2015.
[28] A. Braeken, "Efficient anonym smart card based authentication scheme for multi-server architecture," *International Journal of Smart Home*, vol. 9, no. 9, pp. 177–184, 2015.
[29] F. Wen and D. Guo, "An improved anonymous authentication scheme for telecare medical information systems," *Journal of medical systems*, vol. 38, no. 5, pp. 1–11, 2014.
[30] M.-C. Chuang and M. C. Chen, "An anonymous multi-server authenticated key agreement scheme based on trust computing using smart cards and biometrics," *Expert Systems with Applications*, vol. 41, no. 4, pp. 1411–1418, 2014.
[31] M. Xu, L. Ma, F. Xia, T. Yuan, J. Qian, and M. Shao, "Design and implementation of a wireless sensor network for smart homes," in *Ubiquitous Intelligence & Computing and 7th International Conference on Autonomic & Trusted Computing (UIC/ATC), 2010 7th International Conference on*. IEEE, 2010, pp. 239–243.
[32] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Transactions on information theory*, vol. 29, no. 2, pp. 198–208, 1983.
[33] "AVISPA: Automated Validation of Internet Security Protocols and Applications," http://www.avispa-project.org/web-interface/basic.php.
[34] M. Burrows, M. Abadi, and R. M. Needham, "A logic of authentication," in *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 426, no. 1871. The Royal Society, 1989, pp. 233–271.
[35] A. Armando, D. Basin, Y. Boichut, Y. Chevalier, L. Compagna, J. Cuéllar, P. H. Drielsma, P.-C. Héam, O. Kouchnarenko, J. Mantovani *et al.*, "The avispa tool for the automated validation of internet security protocols and applications," in *International Conference on Computer Aided Verification*. Springer, 2005, pp. 281–285.
[36] M. L. Das, "Two-factor user authentication in wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 8, no. 3, pp. 1086–1090, 2009.
[37] M. Drozda, S. Schaust, and H. Szczerbicka, "Ais for misbehavior detection in wireless sensor networks: Performance and design principles," in *2007 IEEE Congress on Evolutionary Computation*. IEEE, 2007, pp. 3719–3726.
[38] "TelosB datasheet." [Online]. Available: http://www.willow.co.uk/TelosBDatasheet.pdf
[39] "TinyOS Tutorials." [Online]. Available: http://tinyos.stanford.edu/tinyos-wiki/index.php/TinyOSTutorials
[40] J. Lee, K. Kapitanova, and S. H. Son, "The price of security in wireless sensor networks," *Computer Networks*, vol. 54, no. 17, pp. 2967–2978, 2010.
[41] D. Eastlake, 3rd and P. Jones, "Us secure hash algorithm 1 (sha1)," United States, 2001.
[42] A. Liu and P. Ning, "Tinyecc: A configurable library for elliptic curve cryptography in wireless sensor networks," in *Information Processing in Sensor Networks, 2008. IPSN'08. International Conference on*. IEEE, 2008, pp. 245–256.