Computers, Materials & Continua
DOI:10.32604/cmc.2021.018808
Article

A Lightweight Anonymous Device Authentication Scheme for Information-Centric Distribution Feeder Microgrid

Anhao Xiang and Jun Zheng*

Department of Computer Science and Engineering, New Mexico Institute of Mining and Technology, Socorro, 87801, NM, USA
*Corresponding Author: Jun Zheng. Email: jun.zheng@nmt.edu
Received: 22 March 2021; Accepted: 23 April 2021

Abstract: Distribution feeder microgrid (DFM) built based on existing distributed feeder (DF), is a promising solution for modern microgrid. DFM contains a large number of heterogeneous devices that generate heavy network traffice and require a low data delivery latency. The information-centric networking (ICN) paradigm has shown a great potential to address the communication requirements of smart grid. However, the integration of advanced information and communication technologies with DFM make it vulnerable to cyber attacks. Adequate authentication of grid devices is essential for preventing unauthorized accesses to the grid network and defending against cyber attacks. In this paper, we propose a new lightweight anonymous device authentication scheme for DFM supported by named data networking (NDN), a representative implementation of ICN. We perform a security analysis to show that the proposed scheme can provide security features such as mutual authentication, session key agreement, defending against various cyber attacks, anonymity, and resilience against device capture attack. The security of the proposed scheme is also formally verified using the popular AVISPA (Automated Validation of Internet Security Protocols and Applications) tool. The computational and communication costs of the proposed scheme are evaluated. Our results demonstrate that the proposed scheme achieves significantly lower computational, communication and energy costs than other state-of-the-art schemes.

Keywords: Mutual authentication; information-centric networking; named data networking; distribution feeder microgrid; smart devices; AVISPA; security

1 Introduction

Smart grids. provide a more reliable and efficient power supply than traditional power grids by incorporating advanced information and communication technologies (ICT) [1,2]. Microgrids are a subset of smart grids that achieve grid deployment in small regions. A microgrid acts as a single controlled entity that is formed by a group of interconnected load and demand resources with communication and control capabilities [3]. It has a well-defined electricity boundary with a limited number of connection points to the utility grid such that it can operate in either grid-connected or islanded mode.

Distribution feeder microgrid (DFM) has been proposed as a solution of modern microgrid which is built based on existing distribution feeder (DF) [3,4]. DFM utilizes advanced communication, control, and protection technologies to increase the sustainability, reliability, and resiliency of the grid and support very high penetration of distributed energy resources (DERs) [3,5]. The architecture of DFM is illustrated in Fig. 1, which contains a variety of demand and load entities such as consumer appliances, generators, energy storage, electrical vehicles (EVs), DERs, smart meters, synchrophasor devices etc. The DFM gateway (DG) serves as the central control and management entity that connects the DFM to the utility grid.

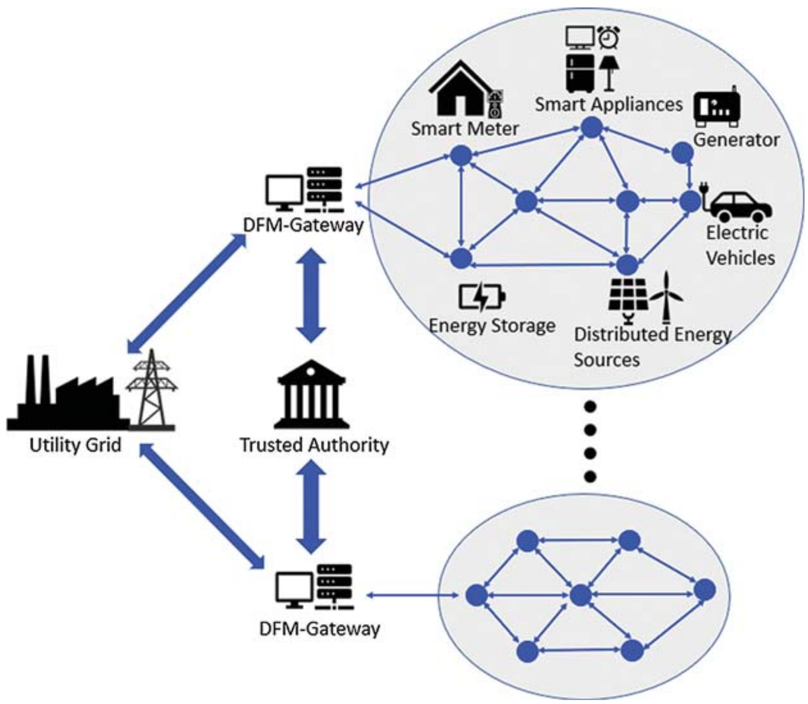

Figure 1: System architecture of DFM

One of the major technical challenges faced by DFM is the communication demand of a large number of heterogeneous devices. A scalable networking and communication architecture is needed that can meet requirements such as low data delivery latency and heavy network traffic [6]. The information centric networking (ICN) paradigm has been explored recently to address the requirements of smart grid communication [6–9]. Unlike the host-centric IP-based networking architecture, ICN adopts a content-centric communication model with novel features like data caching in network edge, data provenance, inherent multicast support, etc. which make it suitable for smart grid applications. C-DAX (Cyber-secure Data and Control Cloud) is an ICN-based solution proposed for the monitoring and control of smart grids [8]. Tourani et al. [6] proposed an ICN-based smart grid networking architecture called iCenS, which was shown to be effective in serving various types of smart grid traffic. Yu et al. [9] proposed a Content-Centric Networking (CCN) based advanced metering system (CCN-AMI) for smart grids. The CCN-AMI system is comprised of several components such as smart meters, demand response management system (DRMS), which provides better traffic congestion control, mobility and cyber security. Ravikumar et al. [7] proposed an ICN-based smart grid architecture that consists of a three-level hierarchy for information flow including physical level, aggregation level, and computation level. The hierarchy specifies constituents and the interaction mechanism at each level. The proposed architecture adopts IEC 61850 as underlying communication stack for backward compatibility and adds the Information-Centric Network Protocol (ICNP) layer. Both work of [7,9] and have conducted a comprehensive performance analysis of the proposed ICN architectures and the results show a great potential of applying ICN for smart grids.

In this paper, we consider a named data networking (NDN) based architecture to address the communication demand of DFM. NDN is a representative ICN architecture which has been shown as a promising solution for not only smart grid communication [6,7] but also the communication needs of applications of smart cities [10], smart campus [11], smart home [12], and smart healthcare [13]. In addition to communication requirements, another key technical challenge faced by DFM is to ensure the security and privacy of the grid. The integration of advanced ICT technologies in DFM makes it vulnerable to a number of cyber attacks such as man-in-the-middle (MITM) attacks, reply attacks, impersonation attacks, etc. Adequate authentication is essential for preventing unauthorized access to the grid network and defending against cyber attacks. There are lots of authentication and key agreement protocols proposed for smart grids based on IP networking architecture. For example, Garg et al. [14] proposed an ECC (Elliptic Curve Cryptography) and FHMQV (Fully Hashed Menezes–Qu-Vanstone) based authentication scheme for smart metering infrastructure (SMI). Kumar et al. [15] proposed another ECC-based authentication scheme for smart grid device and utility center communication. Chen et al. [16] proposed an ECC and bilinear pairing-based authentication scheme for smart grid communication. Zhang et al. [17] proposed a lightweight authentication scheme using symmetric cryptography, hash, and other lightweight operations.

There are some works on authentication protocols designed for ICN-based networking architectures, mainly for supporting various IoT communication scenarios. Similar to IP-based networking architecture, authentication also brings significant security benefits to ICN-based networking architecture [18]. Compagno et al. [18] proposed a secure IoT device onboarding protocol for ICN called OnboardICNg based on symmetric-key cryptography. It was shown in [19] that OnboardICNg incurs significant lower time and energy overheads compared with the design based on asymmetric-key cryptography. LASeR, a secure IoT device authentication and routing scheme for NDN-based smart cities, was proposed in [20]. The device authentication of LASeR is based on the Pre-Shared Key Extensible Authentication Protocol (EAP-PSK). For ICN based DFM, the authentication scheme should provide various security features including mutual authentication, session key agreement, defending against various attacks, anonymity, and resilience against device capture attack [15]. In addition, majority of smart devices in DFM are resource-limited which requires the authentication scheme to have low computational, communication, and energy costs.

The contributions of this paper are: (1) we propose a lightweight anonymous device authentication scheme for NDN-based DFM; (2) we perform an analysis of security requirements satisfied by the proposed scheme and formally verify its security by using the popular AVISPA (Automated Validation of Internet Security Protocols and Applications) tool [21]; and (3) we conduct a performance comparison of the proposed scheme with existing schemes to demonstrate that the proposed scheme achieves lower computational, communication, and energy costs.

The rest of this paper is organized as follows: Section 2 introduces system models and assumptions adopted in this paper. The proposed device authentication scheme for NDN-based DFM is presented in Section 3. In Section 4, we analyze security requirements satisfied by the proposed scheme followed by a formal security verification with the AVISPA tool. The performance of the proposed scheme in terms of computational, communication, and energy costs is evaluated and compared with other state-of-the-art schemes in Section 5. Finally, the conclusion of this paper is drawn in Section 6.

2 System Models and Assumptions

In this section, we introduce the network model of NDN based DFM, the threat model, and their assumptions after an overview of NDN.

2.1 NDN Overview

NDN is a new ICN paradigm proposed as a candidate for future internet architecture. NDN assigns a unique name to a trunk of data or a so-called content object. NDN has two types of packets: *Interest* and *Data* packets. The *Interest* packet is issued by a consumer to request the desired data content using the unique name. The network will forward the *Interest* packet to the provider of the data content. The provider will reply with a *Data* packet back to the consumer which contains the name and actual content of the data. *Interest* and *Data* packets can have other fields besides the name of the data content. In our scheme, we only consider the **name** field in the *Interest* packet, and the **name**, **content**, and **signature** fields in the *Data* packet.

Routing of NDN is done through three data structures maintained by each NDN router: a Pending Interest Table (PIT), a Forwarding Information Base (FIB), and a Content Store (CS). The CS serves as the data cache of an NDN router. When an *Interest* packet arrives, the router will check if the name of the requested data content matches any record in the CS and serves the data if there is a match. Otherwise, the router will check the PIT table to avoid forwarding duplicated *Interest* packet. If no PIT entry can be found, the router will use the FIB table to determine the appropriate interface to forward the *Interest* packet. In the meantime, the PIT table will also be updated to indicate that the *Interest* packet is forwarded. The routing of the corresponding *Data* packet will simply use the reverse path identified in the PIT.

In NDN, a *Data* packet usually contains the name of the corresponding *Interest* packet. This duplication will tremendously increase the size of a *Data* packet when a long name is used for the corresponding *Interest* packet. This causes a significant problem when transmitting an NDN packet over a low power wireless link such as an IEEE 802.15.4 link due to its limited maximum physical packet size. Solutions relying on fragmentation and reassembly [22] could result in a significant increase in memory storage, processing complexity, and traffic amount. In this paper, we adopt a solution proposed in [23] that replaces a long *Interest* name with a short 1-byte HopID. The solution extends the PIT table with two new columns: HID_i and HID_o. For an *Interest* packet, each hop generates a 1-byte HopID and includes it in the name. The HopID will be stored in the HID_o column which should be unique within the local PIT table and has the same lifetime as the corresponding PIT entry. When an *Interest* packet arrives at a hop, the HopID will be extracted from the *Interest* name and stored in the HID_i column of the corresponding PIT entry. A new HopID will then be generated by the hop and stored in the HID_o column of the same PIT entry. The new HopID will be included in the name of the outgoing *Interest* packet. This process will be performed in each intermediate hop until the *Interest* is served by the producer. The producer will extract HopID from the HID_i column and use it as the name of the responded *Data* packet.

Intermediate hops that forward the *Data* packet will simply extract the HopID and lookup HID_o column of the PIT table for a match. If a match is found, the hop will replace the HopID of the *Data* packet with the new HopID from the HID_i column of the matched PIT entry before forwarding the *Data* packet.

2.2 Network Model and Assumptions

We consider that all entities of a DFM shown in Fig. 1 are wirelessly connected to form a mesh network topology. The load and demand entities with communication and control capabilities in a DFM are referred as smart devices. The majority of them have limited computational, memory, and energy resources. Each device has a unique and immutable real identity such as a Silicon-ID number [24]. The deployment of smart devices is done over time. The connection of a DFM to the utility grid is done through the DG, which is considered as resource un-constrained. A smart device in a DFM may connect to the DG through a multi-hop path with the help of other devices. We also assume that a Trust Authority (TA) is existed to serve DFMs of a utility service provider as shown in Fig. 1. The TA provides authentication and authorization services to bootstrap new smart devices into a DFM network.

2.3 Threat Model and Assumptions

The basic adversary model considered for the proposed scheme is the widely used Dolev–Yao (DY) model [25]. According to the model, all entities including smart devices and DG are not trustworthy. The messages between the entities are transferred through an open channel which can be eavesdropped, intercepted, and modified by an adversary. In addition, we assume that an adversary can compromise a session key and session states according to Canetti and Krawczyk (CK) adversary model [26]. The adversary can also physically capture a device to extract the stored secret credentials by using the sophisticated power analysis attacks [27]. Finally, we assume that the TA is a fully trusted entity and can't be compromised.

Based on the threat model and assumptions, the proposed scheme aims to satisfy security requirements including message integrity, mutual authentication and session key agreement, perfect forward secrecy, anonymity, and resistance to various attacks.

3 Proposed Scheme

The proposed scheme consists of two phases: (1) device registration phase; (2) network discovery and authentication phase. Note that the TA is only involved in the device registration phase. Tab. 1 lists the notations and their descriptions used in this paper.

3.1 Device Registration Phase

Before deployed in a DFM, a smart device $S(SD_S)$ needs to be registered offline at the TA by the owner who brings the device to the TA's office to complete the registration through a secure channel [28]. During the registration process, SD_S first sends its real identity ID_S to TA. TA then generates a master secret k_S and two random numbers r_{TA-DG}, and r_{TA-SD_S} for SD_S. The pseudo-identity of SD_S is then computed as $PID_S = H(ID_S||k_S)$. TA also computes two secrets $A_S = H(ID_G||PID_S||r_{TA-DG})$ and $B_S = H(ID_S||PID_G||r_{TA-SD_S})$. Note that ID_G and PID_G are the real identity and pseudo-identity of DG, respectively. Finally, TA sends PID_S, A_S, and r_{TA-SD_S} to SD_S, and then sends PID_S, B_S, and r_{TA-DG} to DG. The device registration phase is illustrated in Fig. 2.

Table 1: Notations and their descriptions used in this paper

Notation	Description
ID_X	Identity of entity X
PID_X	Pseudo-identity of entity X
$HopID_X$	HopID generated by entity X
k_X	Master secret for entity X
r_*	Random number
A_S, B_S	Secrets
SK_{X-Y}	Session key between entities X and Y
$H()$	One-way hash function
$E_K(M)$	Encrypt message M using key K
$D_K(M)$	Decrypt message M using key K
$S_K(M)$	Sign message M using key K
$\|\|$	Message concatenation

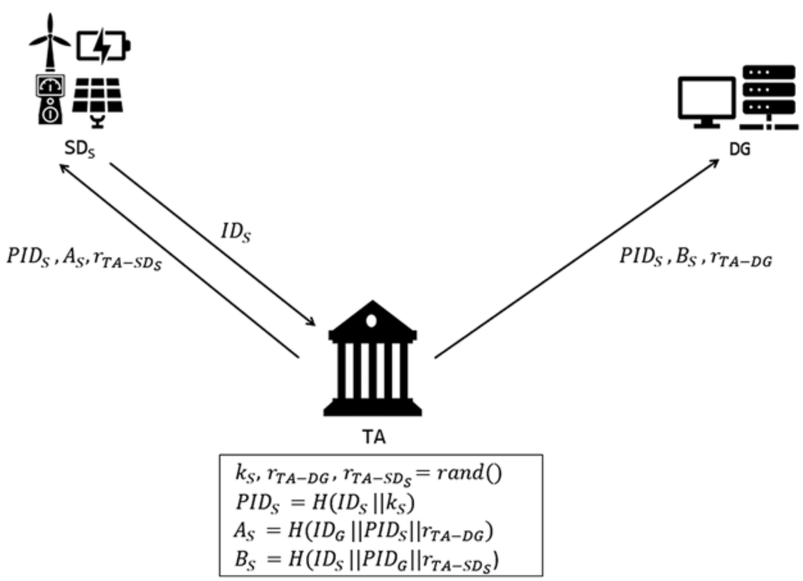

Figure 2: An illustration of device registration phase

3.2 Network Discovery and Authentication Phase

After the registration, SD_S performs the network discovery and authentication phase to join the trusted network of a DFM. The procedure of this phase is illustrated in Fig. 3 and described as follows:

- SD_S generates a random number r_{SD_s-DG} and a HopID $HopID_S$, and then computes $C_1 = E_{A_S}(r_{SD_S-DG})$ and $S_1 = S_{A_S}(PID_S||r_{SD_S-DG})$. After that, SD_S generates an *Interest* with the name as $/Discover/PID_S/C_1/S_1/HopID_S$. A PIT entry will be created with name prefix $/Discover/PID_S/C_1/S_1$ and $HopID_S$ is stored in the HID_ocolumn of this entry. This *Interest* will then be broadcast to all neighbors of SD_S.

- Upon receiving the broadcast *Interest*, a trusted neighbor device $N(SD_N)$ can choose to help the network discovery and authentication process of SD_S or not. If SD_N wants to help the process, it will extract $HopID_S$ and S_1 from the received *Interest*. A PIT entry for the received *Interest* is created with name prefix $/Discover/PID_S/C_1/S_1$ and the HID_i column as $HopID_S$. SD_N then generates a new HopID $HopID_N$ and stores it in the HID_o column of the newly created PIT entry. A signature S_2 will be computed as $S_{SK_{SD_N-DG}}(S_1||PID_G||PID_N)$, where SK_{SD_N-DG} is the session key shared between SD_N and DG, and PID_N is the pseudo-identity of SD_N. Finally, a new *Interest* is generated and sent to DG with the name as $/Auth/PID_G/PID_S/C_1/PID_N/S_2/HopID_N$. Note that a mapping from the new *Interest* name $/Auth/PID_G/PID_S/C_1/PID_N/S_2$ to the broadcast *Interest* name $/Discover/PID_S/C_1/S_1$ must be established at SD_N.

When the new *Interest* is forwarded through the trusted network of the DFM to DG, the HopID part of the *Interest* name will be replaced by a new HopID generated at each hop. Supposing the hop before DG is a smart device $M(SD_M)$ and its generated HopID is $HopID_M$, the name of the *Interest* received by DG will be $/Auth/PID_G/PID_S/C_1/PID_N/S_2/HopID_M$. Without loss of generality, we assume that the *Interest* sent by SD_N will be received by DG directly.

- When DG receives the *Interest*, a PIT entry with the name prefix $/Auth/PID_G/PID_S/C_1/PID_N/S_2$ will be created with the corresponding HID_i set as $HopID_N$. It extracts PID_S^* and C_1^* from the *Interest* name. Then A_S^* is computed as $A_S^* = H(ID_G||PID_S^*||r_{TA-DG})$ which is used to decrypt C_1^* to obtain $r_{SD_s-DG^*} = D_{A_S^*}(C_1^*)$. After that, DG computes $S_1^* = S_{A_S^*}(PID_S^*||r_{SD_S-DG^*})$, and $S_2^* = S_{SK_{SD_N-DG}}(S_1^*||PID_G||PID_N^*)$. It then checks if $S_2^* == S_2$. If not, the authentication process will be aborted. Otherwise, SD_S is authenticated at DG which will then generate two random numbers r_{DG-SD_S} and $r_{SD_s-SD_N}$. The two random numbers are used to generate the session key between SD_S and SD_N as $SK_{SD_S-SD_N} = H(A_S^*||B_S||r_{SD_s-DG^*}||r_{SD_s-SD_N})$ and the session key between SD_S and DG as $SK_{SD_S-DG} = H(A_S^*||B_S||r_{SD_s-DG^*}||r_{DG-SD_S})$. DG will prepare the *Data* packet by computing $C_2 = E_{SK_{SD_N-DG}}(SK_{SD_S-SD_N})$, $C_3 = E_{B_S}(r_{DG-SD_s}||r_{SD_S-SD_N})$, and $S_3 = S_{B_S}(r_{DG-SD_S}||r_{SD_S-SD_N}||PID_G||PID_N)$, which are included as the content. DG will generate a signature for the *Data* packet as $S_4 = S_{SK_{SD_N-DG}}(C_3||S_3||SK_{SD_S-SD_N})$. Then $HopID_N$ is retrieved from the HID_i column of the corresponding PIT entry which will be used as the name of the *Data* packet. The *Data* packet will be sent back to SD_N.
- When SD_N receives the *Data* packet, it first extracts $HopID_N$ from the name and look up the HID_i column of the matched PIT entry to find the next hop's HopID $HopID_S$, which will be used as the name of the new *Data* packet sent back to SD_S. Then SD_N will extract C_2^*, C_3^*, S_3^* from the content of the received *Data* and obtain the session key $SK_{SD_S-SD_N}$ by decrypting C_2^* with SK_{SD_N-DG}. After that, it generates $S_4^* = S_{SK_{SD_N-DG}}(C_3^*||S_3^*||SK_{SD_S-SD_N}^*)$ and verifies if $S_4^* == S_4$. If not, the authentication process will be aborted. Otherwise, SD_N sends a *Data* packet to SD_S whose content includes PID_G, PID_N, and C_3 with the name as $HopID_S$ and the signature as S_3.
- Upon receiving the *Data* packet from SD_N, SD_S first computes $B_S^* = H(ID_S||PID_G^*||r_{TA-SD_S})$ and obtains $r_{DG-SD_S}^*$ and $r_{SD_S-SD_N}^*$ by decrypting C_3^* with B_S^*. Then SD_S computes $S_3^* = S_{B_S^*}(r_{DG-SD_S}^*||r_{SD_S-SD_N}^*||PID_G^*||PID_N^*)$ and verifies if $S_3^* == S_3$. If not, the authentication process will be aborted. Otherwise, SD_S authenticates DG as legitimate and

computes the two session keys $SK_{SD_S-SD_N} = H(A_S||B_S^*||r_{SD_s-DG}||r^*_{SD_s-SD_N})$, and $SK_{SD_S-DG} = H(A_S||B_S^*||r_{SD_s-DG}||r^*_{DG-SD_S})$.

Note that there could be multiple neighboring devices helping the authentication of SD_S. For *Interest* packets received from different neighboring devices, DG will keep using the same r_{DG-SD_S} so that the session key between SD_S and DG remains the same. DG will generate different $r_{SD_S-SD_N}$ for neighboring devices so that the session keys between SD_S and neighboring devices are different.

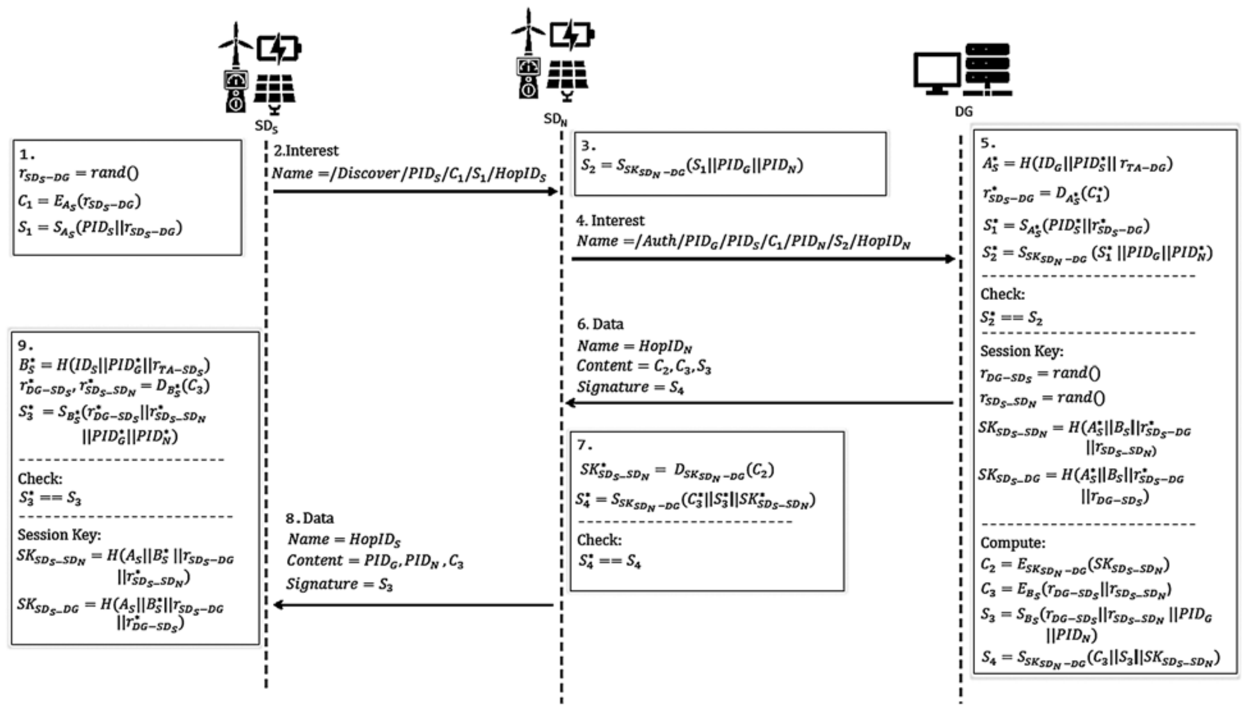

Figure 3: An illustration of network discovery and authentication phase

4 Security Analysis

In this section, we perform an analysis of security requirements satisfied by the proposed scheme and formally verify its security by using the AVISPA tool.

4.1 Informal Security Analysis

Based on the threat model specified in Section 2.3, the proposed scheme can satisfy the following security requirements.

1) Message integrity: The proposed scheme generates a message signature by using the AES-CMAC algorithm to ensure message integrity. Secrets A_S, B_S and secure session key $SK_{SD_{SN}-DG}$ are used as keys for the AES-CMAC algorithm. Since an adversary can't obtain these cryptographic materials from intercepted messages, they can't forge a legitimate message signature after modifying a message.

2) Mutual authentication and session key agreement: Mutual authentication is performed to verify the legitimacy of participating entities. In the proposed scheme, the mutual authentication between SD_S and DG is achieved by using secrets A_S and B_S. DG authenticates SD_S by verifying S_2^* with secret A_S and session key $SK_{SD_{SN}-DG}$. Similarly, SD_S authenticates DG by verifying S_3^* with secret B_S.

In the proposed scheme, after performing mutual authentication for a session, a symmetric session key is established between SD_S and DG as $SK_{SD_S-DG} = H(A_S||B_S||r_{SD_s-DG}||r_{DG-SD_S})$, which can be used to encrypt subsequent communication. Similarly, a symmetric session key between SD_S and its neighbor SD_N is established as $SK_{SD_S-SD_N} = H(A_S||B_S||r_{SD_s-DG}||r_{SD_S-SD_N})$, which can be used to support secure communication between neighboring devices.

3) Perfect forward secrecy: Perfect forward secrecy ensures that the compromising of long-term secret information of legitimate entities (smart devices and DG) by an adversary should not compromise the session keys established in previous sessions. The proposed scheme generates three random numbers r_{SD_s-DG}, r_{DG-SD_S}, and $r_{SD_S-SD_N}$ to compute the two session keys SK_{SD_S-DG} and $SK_{SD_S-SD_N}$ in each session. Without knowing the random numbers, the adversary can't obtain the session keys of previous sessions. Thus, perfect forward secrecy is held by the proposed scheme.

4) Anonymity: Anonymity ensures that the real identity of an entity can't be revealed by an adversary through intercepted messages. The proposed scheme uses a pseudo-identity for each entity that is computed from the real identity and a master secret generated by the TA. It's infeasible for an adversary to compute the real identity without the knowledge of the master secret. Thus, anonymity is satisfied by the proposed scheme.

5) Resistance to impersonation attacks: We consider three cases of impersonation attacks for the proposed scheme:

- *New device impersonation attack*: To impersonate a legitimate new smart device SD_S, an adversary needs to generate a valid *Interest* as the network discovery and authentication request broadcast to neighboring devices. However, the adversary doesn't have the knowledge of A_S to compute C_1 and S_1 to generate a valid *Interest*. Thus, the proposed scheme can resist the new device impersonation attack.
- *Neighboring device impersonation attack*: To impersonate a legitimate neighboring device, an adversary needs to generate a valid *Interest* sent to DG. However, the adversary doesn't have the knowledge of $SK_{SD_{SN}-DG}$ to compute S_2 to generate a valid *Interest*. Thus, the proposed scheme can resist the neighboring device impersonation attack.
- *DG impersonation attack*: To impersonate a legitimate DG, an adversary needs to interpret a received *Interest* and generate a valid *Data* as the response which is impossible since the adversary doesn't have the knowledge of A_S and B_S. Thus, it's infeasible for an adversary to launch the DG impersonation attack.

6) Resistance to replay attacks: An adversary can intercept the transmitted messages and reply them back in a later time. In the proposed scheme, the adversary can't generate the session keys from the intercepted messages. To generate the session keys, the adversary needs to know A_S and B_S, and the three random numbers r_{SD_s-DG}, r_{DG-SD_S}, and $r_{SD_S-SD_N}$ which can't be obtained from the intercepted messages. Therefore, the proposed scheme can resist replay attacks.

7) Resistance of MITM attacks: An adversary can launch MITM attacks by intercepting the transmitted messages and try to make two legitimate entities believe that they communicate with each other directly. To make this happen, the adversary has to know A_S and B_S, or SK_{SD_N-DG}

which are infeasible to be obtained from the intercepted messages. Thus, the proposed scheme can resist MITM attacks.

8) Resilience against devices capture attack: A smart device deployed in the wild could be physically captured by an adversary. Based on the threat model discussed in Section 2.3, the adversary can obtain the secret credentials for authentication such as PID_S, A_S, and B_S from a stolen device by using the power analysis attacks [27]. Such side-channel attacks are difficult to defend unless the device is tamper-resistant [29]. However, the computation of the secret credentials such as A_S and B_S involves ID_S, a unique and immutable identity, so that they are distinct for all smart devices in the DFM network. Thus, the adversary can't compute the session keys between DG and other non-compromising devices using the secret credentials of the captured device. Such security property is called unconditional security against device capture attack [15,24,30–32]. Therefore, the proposed scheme is resilient against device capture attack.

4.2 Formal Security Verification

In this section, we formally verify the security of the proposed scheme by using the AVISPA tool, which is designed for the analysis of large-scale internet security-sensitive protocols [21].

In AVISPA, the protocol actions and security requirements are described with a language called the High-Level Protocol Specification Language (HLPSL). AVISPA generates an intermediate file (IF) from the input HLPSL file by using the HLPSL2IF translator and passes the intermediate file to an AVISPA backend. The backend will verify the protocol security and generates a security report. AVISPA has four different backends: On-the-fly Model-Checker (OFMC), CL-based Attack Searcher (CL-AtSe), SAT-based Model-Check (SATMC), and Tree Automata-based Protocol Analyzer (TA4SP). User can choose suitable backends for protocol security verification.

HLPSL is a role-based language that contains two types of roles: basic role and composition role. Figs. 4–6 describe the initial parameters, states, and transitions for the three basic roles (SD_S, SD_N, and DG) involved in the authentication process. The composition roles are specified in Fig. 7. The session role instantiates the parameters of the basic roles. The environment role contains the global variables and specifies the sessions of the protocol. Finally, the security goals of the proposed scheme are also specified in Fig. 7, which test the strength of session keys against various attacks and verify the establishment of mutual authentication. Fig. 8 shows the outputs of the OFMC and CL-AtSe backends, which prove the proposed scheme is safe against both backends.

5 Performance Analysis

In the following sections, we evaluate the communication, computation, and energy costs of the proposed scheme and compare them with those of OnboardICNg [18] and LASeR [20]. OnboardICNg and LASeR adopt similar system architectures as the proposed scheme. Tab. 2 shows the mapping of the entities of OnboardICNg and LASeR to those of the proposed scheme. Since DG is resource-unconstrained, our analysis concentrates on resource-limited smart devices. We assume that there are n neighbor devices helping the authentication process.

5.1 Communication Cost

In this section, we evaluate the communication cost of the proposed scheme during the network discovery and authentication phase in terms of the number of exchanged messages and the number of bytes sent and received by smart devices. We use IEEE 802.15.4 as the underlying link-layer which has a maximum frame size of 127 bytes.

```
role device(
 D,N,G : agent,
 H: hash_func,
 As,Bs: symmetric_key,
 SND,RCV: channel(dy)
)
played_by D def=
local
 State: nat,
 SK1,SK2: message,
 Rsd_dg,Rsd_sdn,Rdg_sd: text
 init State := 0
 transition
 1.State = 0 /\ RCV(start) =|>
  Rsd_dg' := new()
  /\ SND(D.N.{Rsd_dg'}_As)
  /\ witness(D,G,device_As,As)
  /\ State' := 7
 2. State = 7 /\ RCV(N.D.{Rdg_sd'.Rsd_sdn'}_Bs) =|>
   State' := 8 /\  SK1' := H(As.Bs.Rsd_dg.Rsd_sdn')
      /\ SK2' := H(As.Bs.Rsd_dg.Rdg_sd')
      /\ secret(SK1', sessionkey1, {D,G})
      /\ secret(SK2', sessionkey2, {D,N,G})
      /\ request(D,G,gateway_Bs,Bs)
      /\ request(D,G,dg_rdg_sd,Rdg_sd')
      /\ request(D,G,dg_rsd_sdn,Rsd_sdn')
end role
```

Figure 4: Specification of the SD_S role

```
role neighbor(
 D,N,G : agent,
 H: hash_func,
 Kn: symmetric_key,
 SND,RCV: channel(dy)
)
played_by N def=
local
 State: nat,
 SK1,SK2: message,
 X : {text}_symmetric_key,
 Y : {text.text}_symmetric_key
 init State := 2
 transition
 1.State = 2 /\ RCV(D.N.X') =|>
   SND(N.G.{X'}_Kn)
   /\  State' := 3
 2. State = 3 /\ RCV(G.N.{SK1'}_Kn.Y') =|>
   State' := 4 /\ SND(N.D.Y')
end role
```

Figure 5: Specification of the SD_N role

```
role gateway(
 D,N,G : agent,
 H: hash_func,
 As,Bs,Kn: symmetric_key,
 SND,RCV: channel(dy)
)
played_by G def=
local
 State: nat,
 SK1,SK2: message,
 Rsd_dg,Rsd_sdn,Rdg_sd: text,
 PIDs: text
 init State := 5
 transition
 1.State = 5 /\ RCV(N.G.{{Rsd_dg'}_As}_Kn) =|>
   State' := 6
   /\ Rdg_sd' := new()
   /\ Rsd_sdn' := new()
   /\ SK1' := H(As.Bs.Rsd_dg'.Rsd_sdn')
   /\ SK2' := H(As.Bs.Rsd_dg'.Rdg_sd')
   /\ SND(G.N.{SK1'}_Kn.{Rdg_sd'.Rsd_sdn'}_Bs)
   /\ witness(G,D,dg_rdg_sd,Rdg_sd')
   /\ witness(G,D,dg_rsd_sdn,Rsd_sdn')
   /\ witness(G,D,gateway_Bs,Bs)
   /\ secret(SK2', sessionkey2, {D,G})
   /\ request(G,D,device_As,As)
   end role
```

Figure 6: Specification of the ***DG*** role

Since the communication between SD_S and SD_N is untrusted during the authentication process, an 802.15.4 frame exchanged between SD_S and SD_N does not carry the signature which results in a size of 36 bytes for the header and footer. On the other hand, a frame exchanged within the trusted network of DFM requires the full 52-byte 802.15.4 header and footer. In addition, we consider the $1+0$ encoding proposed for NDN packets [33]. Tab. 3 shows the fields and their corresponding sizes for NDN *Interest* and *Data* packets, where S_T is the total size of name components TL (1B * number of name components), S_N is the total size of the name values, and S_C is the total size of the content. We assume that ID and PID are 4 bytes, a random number is 8 bytes, and outputs of electric signature, hash, and encryption operations are 16 bytes. Prefixes (*/Discover* and */Auth*) are encoded in 1 byte. Based on the above assumptions, we compare the communication cost of the proposed scheme with those of OnboardICNg and LASeR in Tab. 4. For the two reference schemes, we compute the number of bytes sent and received by smart devices with and without HopID implemented. It can be seen that HopID can significantly reduce the communication overheads of the reference schemes, especially for LASeR which also has long *Interest* names. Overall, the results show that the proposed scheme is significantly lightweight than the two reference schemes in terms of the number of exchanged messages and the number of bytes sent/received by smart devices.

```
role session(
 D,N,G : agent,
 H: hash_func,
 As,Bs,Kn: symmetric_key
)
def=
 local SD,SN,SG,RD,RN,RG : channel(dy)
 composition
 device(D,N,G,H,As,Bs,SD,RD)
 /\ neighbor(D,N,G,H,Kn,SN,RN)
 /\ gateway(D,N,G,H,As,Bs,Kn,SG,RG)
end role


role environment()
def=
 const d,n,g : agent,
 h: hash_func,
 as,bs,kn,ai,bi,ki : symmetric_key,
 sessionkey1,sessionkey2,gateway_Bs ,device_As: protocol_id,
 dg_rdg_sd,dg_rsd_sdn: protocol_id,
 intruder_knowledge = {d,n,g,h,ai,bi,ki}
 composition
  session(d,n,g,h,as,bs,kn)
  /\session(i,n,g,h,ai,bi,ki)
end role

goal
secrecy_of sessionkey1
secrecy_of sessionkey2
authentication_on gateway_Bs
authentication_on device_As
%authentication_on dg_rdg_sd
%authentication_on dg_rsd_sdn
end goal
```

Figure 7: Specification of the ***Environment*** and ***Session*** role

5.2 Computational Cost

Tab. 5 compares the cryptographic operations performed by the proposed scheme with those of OnboardICNg and LASeR. In the table, 'T_H', 'T_E', 'T_D', 'T_M', and 'T_{HM}' represent execution times of operations of hash. AES-128 encryption and decryption, AES-CMAC, and HMAC, respectively. To measure the computation times of cryptographic operations, we used a Raspberry Pi 3 board as the smart device running OpenSSL C programming language libraries. The measured computation times of AES-128 encryption, AES-128 decryption, SHA-256, AES-CMAC, and HMAC are 4.36 μs, 4.47 μs, 2.69 μs, 5.54 μs, and 10.9 μs, respectively. We then compared the computation time of the proposed scheme with those of OnboardICNg and LASeR. As shown in Tab. 5, both the proposed scheme and LASeR are more computationally efficient than OnboardICNg. The new joining device of the proposed scheme has a lower computational time than that of LASeR when n is less than 18. Note that LASeR does not establish session keys between the new joining device and its neighbor devices.

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/DFM.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 2.19s
  visitedNodes: 792 nodes
  depth: 8 plies
```

```
% CL-AtSe
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL
PROTOCOL
  /home/span/DFM.if
GOAL
  As Specified
BACKEND
  CL-AtSe
STATISTICS
  Analysed   : 18 states
  Reachable  : 6  states
  Translation: 0.01 seconds
  Computation: 0.00 seconds
```

Figure 8: Outputs of OFMC and CL-AtSe backends

Table 2: Mapping of entities in different schemes

Scheme	New device	Neighbor device	Gateway	Trust authority
Proposed scheme	SD_S	SD_N	DG	TA
OnboardICNg [18]	d_j	d_{nbr}	AGW	AAM
LASeR [20]	SN_2	SN_1	AN	IM

Table 3: NDN *Interest* (I) and *Data* (D) packets

Field	Size	***I***	***D***
Packet type TL	1B	√	√
Name TL	1B	√	√
Name component TLVs	$S_T + S_N$	√	√
Content TLV	1B (TL) $+S_C$		√
Signature info TL	1B		√
Signature type TLV	1B (TL) + 1B (V)		√
Signature value TLV	1B (TL) + 16B (V)		√

5.3 Energy Cost

We estimated the computational energy cost by using the formula $E = V * I * t$, where V is the voltage of the input power, I is the current of the circuit, and t is the computation time. Both V and I were obtained from the Raspberry Pi data sheet [34,35]. We estimated the communication energy cost by using the energy cost of sending and receiving one bit on the Raspberry Pi, which was measured as 0.029 μJ and 0.033 μJ, respectively. Fig. 9 compares the energy costs of a new joining device of the three schemes under different number of neighbor devices. Note that the communication costs of OnboardICNg and LASeR in Fig. 9 were estimated with HopID implemented for a fair comparison. The results show that the proposed scheme is more energy-friendly than the two reference schemes.

Table 4: Comparison of communication costs

Scheme	Communi-cation path	No. of messages	No. of bytes sent/received (without HopID)	No. of bytes sent/received (with HopID)
OnboardICNg [18]	d_j/d_{nbr}	7	$46+147$n/306n	$48+133$n/290n
	d_{nbr}/AGW	2	81/226	83/109
LASeR [20]	SN_2/SN_1	4	$146/127+110n$	$150/77+94$n
	SN_1/IM	4	210/961	214/203
Proposed scheme	SD_S/SD_N	2	–	81/85n
	SD_N/DG	2	–	107/125

Table 5: Comparison of computation costs

Scheme	Entity	Cryptographic operations	Computation time (μ*s*)
OnboardICNg [18]	d_j	$2nT_D+8nT_M$	53.26n
	d_{nbr}	$2T_E+T_D+7T_M$	51.97
LASeR [20]	SN_2	$T_D+(4+n)\,T_{HM}$	48.07 + 10.9n
	SN_1	T_{HM}	10.9
Proposed scheme	SD_S	$(2+n)\,T_H+T_E+nT_D+(1+n)\,T_M$	15.28 + 12.7n
	SD_N	T_D+2T_M	15.55

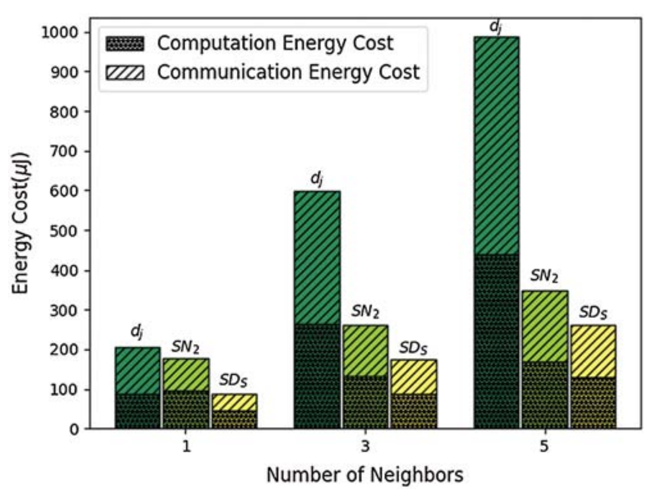

Figure 9: Comparison of energy costs (d_j: OnboardICNg [18], SN_2: LASeR [20], SD_S: proposed scheme)

6 Conclusion

In this paper, we propose a new lightweight anonymous device authentication scheme for NDN-based DFM. We perform an informal analysis of security requirements satisfied by the proposed scheme. Formal security verification of the proposed is also carried out by using the popular AVISPA tool. We conduct a performance evaluation to compare the computational, communication, and energy costs of the proposed scheme with those of other schemes. The results of our security analysis and performance evaluation reveal that the proposed scheme has lower computational and communication overheads than other state-of-the-art schemes. In future, we plan to develop an efficient group key agreement scheme for smart devices in information-centric DMF. We will also research how to perform secure and reliable access control of smart devices in information-centric DMF.

Funding Statement: This material is based upon work funded by the National Science Foundation EPSCoR Cooperative Agreement OIA-1757207.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] S. Lakshminarayana, A. Kammoun, M. Debbah and H. V. Poor, "Data driven false data injection attacks against power grid: A random matrix approach," *arXiv preprint*, vol. 2002.02519, pp. 635–646, 2020.

[2] O. M. Butt, M. Zulqarnain and T. M. Butt, "Recent advancement in smart grid technology: Future prospects in the electrical power network," *Ain Shams Engineering Journal*, vol. 12, no. 1, pp. 687–695, 2021.

[3] J. Wang, L. M. Costa and B. M. Cisse, "From distribution feeder to microgrid: An insight on opportunities and challenges," in *Proc. of IEEE Int. Conf. on Power System Technology*, Wollongong, Australia, pp. 1–6, 2016.

[4] Y. Yan, Y. Qian, H. Sharif and D. Tipper, "A survey on smart grid communication infrastructures: Motivations, requirements and challenges," *IEEE Communications Surveys and Tutorials*, vol. 15, no. 1, pp. 5–20, 2013.

[5] J. Lee, J. P. Berard, G. Razeghi and S. Samuelsen, "Maximizing PV hosting capacity of distribution feeder microgrid," *Applied Energy*, vol. 2020, pp. 261, 2020.

[6] R. Tourani, S. Misra, T. Mick, S. Brahma, M. Biswal *et al.,* "iCenS: An information-centric smart grid network architecture," in *Proc. of 2016 IEEE Int. Conf. on Smart Grid Communications*, Sydney, Australia, pp. 417–422, 2016.

[7] G. Ravikumar, D. Ameme, S. Misra, S. Brahma, R. Tourani *et al.,* "An information-centric network architecture for wide area measurement systems," *IEEE Transactions on Smart Grid*, vol. 11, no. 4, pp. 3418–3427, 2020.

[8] W. K. Chai, N. Wang, K. V. Katsaros, G. Kamel, G. Pavlou *et al.,* "An information-centric communication infrastructure for real-time state estimation of active distribution networks," *IEEE Trans. on Smart Grid*, vol. 6, no. 4, pp. 2134–2146, 2015.

[9] K. Yu, L. Zhu, Z. Wen, A. Mohammad, Z. Zhou *et al.,* "CCNAMI: Performance evaluation of content-centric networking approach for advanced metering infrastructure in smart grid," in *Proc. of 2014 IEEE Int. Workshop on Applied Measurements for Power Systems*, Aachen, French, pp. 1–6, 2014.

[10] S. H. Bouk, S. H. Ahmed, D. Kim and H. Song, "Named-datanetworking-based ITS for smart cities," *IEEE Communications Magazine*, vol. 55, no. 1, pp. 105–111, 2017.

[11] S. Arshad, M. A. Azam, S. H. Ahmed and J. Loo, "Towards information-centric networking (ICN) naming for internet of things (IoT): The case of smart campus," in *Proc. of the Int. Conf. on Future Networks and Distributed Systems*, Cambridge, UK, pp. 1–6, 2017.
[12] M. Amadeo, C. Campolo, A. Iera and A. Molinaro, "Information centric networking in IoT scenarios: The case of a smart home," in *Proc. of 2015 IEEE Int. Conf. on Communications*, London, UK, pp. 648–653, 2015.
[13] M. Chen, "NDNC-BAN: Supporting rich media healthcare services via named data networking in cloud-assisted wireless body area networks," *Information Sciences*, vol. 284, pp. 142–156, 2014.
[14] S. Garg, K. Kaur, G. Kaddoum, J. J. P. C. Rodrigues and M. Guizani, "Secure and lightweight authentication scheme for smart metering infrastructure in smart grid," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 5, pp. 3548–3557, 2020.
[15] N. Kumar, G. S. Aujla, A. K. Das and M. Conti, "ECCAuth: A secure authentication protocol for demand response management in a smart grid system," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 12, pp. 6572–6582, 2019.
[16] Y. Chen, J. Martınez, P. Castillejo and L. Lopez, "A bilinear map pairing based authentication scheme for smart grid communications: PAuth," *IEEE Access*, vol. 7, pp. 22633–22643, 2019.
[17] L. Zhang, L. Zhao, S. Yin, C. H. Chi, R. Liu *et al.,* "A lightweight authentication scheme with privacy protection for smart grid communications," *Future Generation Computer Systems*, vol. 100, no. 9, pp. 770–778, 2019.
[18] A. Compagno, M. Conti and R. Droms, "OnboardICNg: A secure protocol for on-boarding IoT devices in ICN," in *Proc. of the 3rd ACM Conf. on Information-Centric Networking*, Kyoto, Japan, pp. 166–175, 2016.
[19] M. Enguehard, R. Droms and D. Rossi, "On the cost of secure association of information centric things," in *Proc. of the 3rd ACM Conf. on Information-Centric Networking*, Kyoto, Japan, pp. 207–208, 2016.
[20] T. Mick, R. Tourani and S. Misra, "LASeR: Lightweight authentication and secured routing for NDN IoT in control cities," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 755–764, 2018.
[21] L. Vigano, "Automated security protocol analysis with the AVISPA tool," *Electronic Notes in Theoretical Computer Science*, vol. 155, no. 1, pp. 61–86, 2006.
[22] J. Shi and B. Zhang, "NDNLP: A link protocol for NDN, NDN Team, NDN," Technical Report NDN-0006, 2012.
[23] C. Gundogan, P. Kietzmann, T. C. Schmidt and M. Wahlisch, "ICNLoWPAN–named-data networking for low power IoT networks," in *Proc. of 2019 IFIP Networking Conf.*, Warsaw, Poland, pp. 1–9, 2019.
[24] 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.
[25] 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.
[26] R. Canetti and H. Krawczyk, "Analysis of key-exchange protocols and their use for building secure channels," in *Proc. of Int. Conf. on the Theory and Applications of Cryptographic Techniques*, Innsbruck, Austria, pp. 453–474, 2001.
[27] T. S. Messerges, E. A. Dabbish and R. H. Sloan, "Examining smartcard security under the threat of power analysis attacks," *IEEE Transactions on Computers*, vol. 51, no. 5, pp. 541–552, 2002.
[28] M. Nabeel, S. Kerr, X. Ding and E. Bertino, "Authentication and key management for advanced metering infrastructures utilizing physically unclonable functions," in *2012 IEEE Third Int. Conf. on Smart Grid Communications*, Tainan, pp. 324–329, 2012.
[29] S. Dey and A. Hossain, "Session-key establishment and authentication in a smart home network using public key cryptography," *IEEE Sensors Letters*, vol. 3, no. 4, pp. 1–4, 2019.
[30] A. K. Das, M. Wazid, A. R. Yannam, J. J. P. C. Rodrigues and Y. Park, "Provably secure ECC-based device access control and key agreement protocol for IoT environment," *IEEE Access*, vol. 7, pp. 55382–55397, 2019.

[31] M. Wazid, A. K. Das, V. Odelu, N. Kumar and W. Susilo, “Secure remote user authenticated key establishment protocol for smart home environment,” *IEEE Trans. on Dependable and Secure Computing*, vol. 17, no. 2, pp. 391–406, 2020.

[32] J. Srinivas, A. K. Das, M. Wazid and N. Kumar, “Anonymous lightweight chaotic map-based authenticated key agreement protocol for industrial internet of things,” *IEEE Trans. on Dependable and Secure Computing*, vol. 17, no. 6, pp. 1133–1146, 2020.

[33] M. Enguehard, R. E. Droms and D. Rossi, “On the cost of geographic forwarding for information-centric things,” *IEEE Transactions on Green Communications and Networking*, vol. 2, no. 4, pp. 1150–1163, 2018.

[34] Power supply, https://www.raspberrypi.org/documentation/hardware/raspberrypi/power/README.md, (accessed on July 10, 2020).

[35] Raspberry Pi 3 model B+, https://static.raspberrypi.org/files/product-briefs/Raspberry-Pi-Model-Bplus-Product-Brief.pdf, (accessed on July 10, 2020).