A Privacy Preserving and Secure Authentication Protocol for the Advanced Metering Infrastructure with Non-Repudiation Service

Chakib Bekara, Thomas Luckenbach
RESCON Department
Fraunhofer FOKUS Institute
Berlin, Germany
{*chakib.bekara, thomas.luckenbach*}*@fokus.fraunhofer.de*

Kheira Bekara
LOR Department
Institut TELECOM Sud-Paris
Evry, FRANCE
kheira.bekara@it-sudparis.eu

***Abstract*—In the smart grid, smart meters play an important role in keeping a real-time balance between energy production and energy consumption. The advanced metering infrastructure is responsible of collecting, storing, analyzing and providing metering data from smart meters to the authorized parties, and also carrying commands, requests, messages and software updates from the authorized parties to the smart meters. As such, advanced metering infrastructure is one of the important components of the smart grid, and securing it is a prerequisite for guaranteeing a large acceptance and deployment of the smart grid. One step towards securing it is to provide authentication, integrity and confidentiality services. Another important step is to preserve the privacy of the end-customer equipped with a smart meter, where all its related-data (including metering data) are kept secret, such as energy consumption, billing and which smart appliances are used in its premises. In this paper, we propose an ID-based authentication protocol for the advanced metering infrastructure, which provides source authentication, data integrity and non-repudiation services, while preserving the end-customer's privacy.**

Keywords-Identity-based Cryptography*; *Key Establishment*; *Data Source Authentication*; *Smart Grid*; *Advanced Metering Infrastructure*; *Smart Meter*; *User's privacy

I. INTRODUCTION

The term SM (Smart Meter) designs an advanced digital utility meter (electricity, gas, water, heat, etc.) equipped with a two-way communication interface. In the context of the SG (Smart power Grid), a SM is installed at an end-customer premises, and is able to communicate with the utility (energy utility), by sending metering data (e.g., energy consumed/locally produced, grid status, meter status, etc.), and also by responding to messages from the utility (e.g., software update, realtime pricing, load shedding, energy cut-off, etc.).

The SM is a key element in the AMI (Advanced Metering Infrastructure), as shown in Figure 1. The AMI [1] is responsible for collecting, analyzing, storing and providing the metering data sent by the SMs to the appropriate authorized parties (e.g., energy provider, utility, SG operator, etc.). The AMI is also responsible for transmitting requests, commands, pricing-information and software updates from the authorized parties to the SMs. Figure1 presents a simplified view of the AMI as a part of the SG, where we differentiate the following components[2]:

- End-customer's HAN: this includes the SM, which is the main component, in addition to SAs (smart appliances), e-car (electric car) and local renewable energy sources in a customer's premises.
- GW (Gateway): the GW acts as an interface between a set of SMs and the AMI head-end. The GW plays the role of a concentrator, collecting data from several SMs using a local short-distance communication infrastructure (e.g., ZigBee, Bluetooth, WiFi, PLC, etc.), then sending them using a long-distance communication infrastructure (e.g., Internet, GSM, WiMax, GPRS) to the AMI head-end. It could also play the role of a firewall, by protecting the end-customers HAN from outside attackers. There may be several levels of GWs: a GW per a residential building block (a set of HANs), an upper level GW per a set of building blocks, etc.
- AMI Communication Infrastructure: in its simplest form, this network provides a communication path from the SM/GW to the AMI head-end, and reciprocally.
- AMI head-end: is responsible for two-way communication with SMs/GWs. Thus it serves as a gateway between the end-customer's HAN (mainly the SM) and the AMI back-end.
- AMI back-end: the components in the AMI back-end manage SMs, use data collected from SMs (for billing, grid status estimation, consumption/production forecasting, outage management, etc.) and send data and requests (software/firmeware update, real-time pricing, load shedding, etc.) to SMs. The MDMS (Meter Data Management System) [2] is responsible for storing all the exchanged data with the SMs, then dispatching them to the authorized receivers.

The well-functioning of the SG is based on the trustworthy of the overall data flow(assuming authentic origin/source and data integrity), including the data exchanged in the the AMI [3]. Attacking the AMI will impact the utility and the end-customer, and, as consequence, threaten the

 ISBN: 978-1-61208-189-2

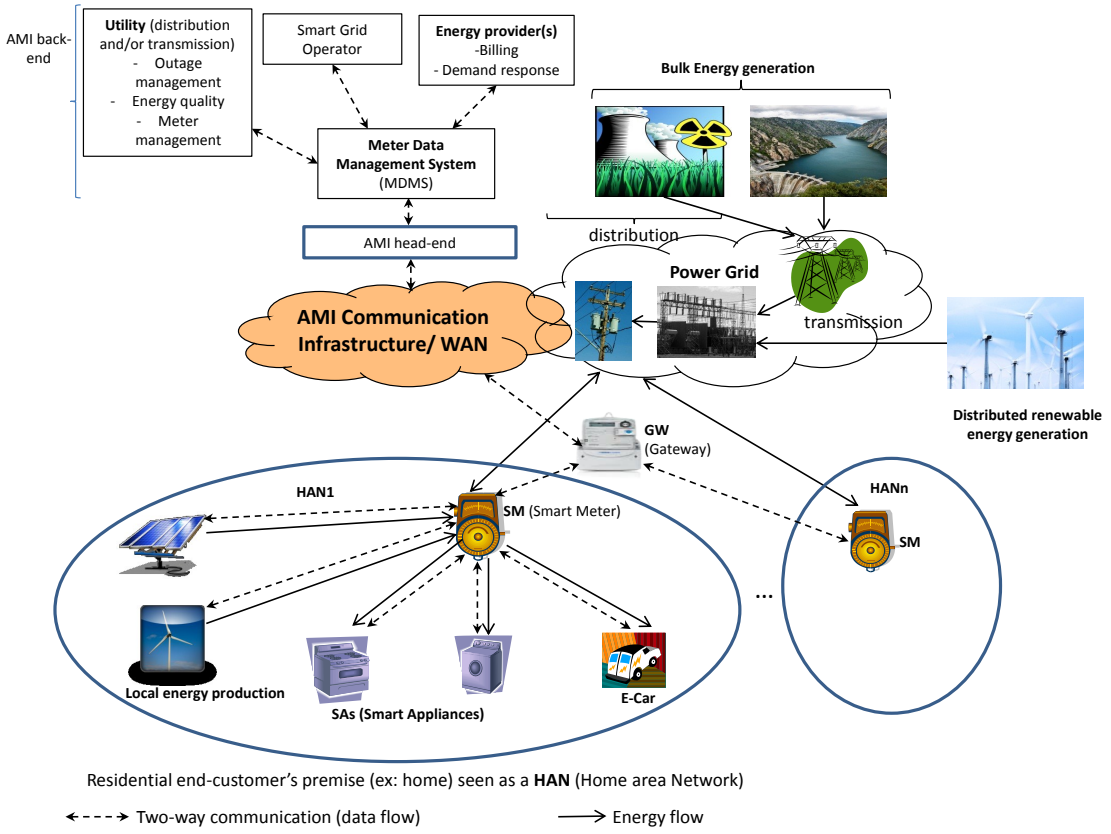

Figure 1. A general view of the AMI as part of the SG

security of the SG. An attacker could impersonate the SMs and send false meter readings to the utility. This may results on financial loss for the utility, unbalanced energy production/consumption and may lead to power outages. Moreover, an attacker could target the end-customers HAN, by masquerading as the utility and sending fake commands (e.g., disconnection or load shedding), fake pricing messages (low price during peak period and high price during off-peak period), resulting on financial loss for the end-customer. The attacker could also impersonate the GW to which is connected the SM and thus drop all messages sent from/to the SM, impacting both the utility and the end-customer. Finally, the attacker could also impersonate the SM to the local SAs inside the end-customers HAN, making them to operate during peak period .

Securing the flow of data in the AMI requires establishing secure (end-to-end) communication channels between communicating parties in the AMI, providing the basic security services: data source authentication, non-repudiation and confidentiality [3], where:

1. Data source authentication allows the verification of the identity that one party claim to have (origin authentication), and that the data received from that party was not altered en-route (data integrity). This service protects the AMI from the following attacks: identity impersonation, MITM (Man-In-The-Middle) [4] attack and data injection/modification.
2. Non-repudiation prevents a sender from denying sending a message. This service is required to avoid that a SM (end-customer) deny sending some metering data (energy consumption), or the energy provider deny sending some real-time pricing. This service is useful when responsibilities in case of dispute need to be clearly identified.
3. Confidentiality protects data from being legible to unauthorized parties.

In our paper, we mainly investigate how to efficiently provide the two first security services. In addition, we consider a third security service, which is related to the end-customer's HAN privacy. Information about the SAs inside the end-customer's HAN (e.g., identity, type, etc.) should not be divulged, even when a SM and a SA need to mutually authenticate.

In this paper, we present an ID(Identity)-based authen-

tication protocol for the AMI, providing data source authentication and non-repudiation services, and preserving the privacy of the end-customer. In Section II, we review some related works about authentication in AMI. In Section III, we describe our motivation towards our proposal, give the assumptions we made and summarize the notations we use. Section IV presents our ID-based authentication protocol for the AMI, and Section V discusses its security and gives a brief comparison with the related works. Section VI gives some perspective works and concludes the paper.

II. Related Works

Several authentication protocols (APs) were proposed for the SG and the AMI, all of them rely on the use of one or more of the following cryptographic keying materials/schemes:

1. Using traditional PKC (public-key cryptography); coupled with a PKI [5] (public key infrastructure), as in [6].
2. Using IBC (ID-based Cryptography) [7]; coupled with a single trusted PKG (Private Key Generator), as in [8].
3. Using solely symmetric-key cryptography; coupled with a single trusted KDC (Key Distribution Center), as in [9] [10].

The proposed authentication protocols achieve authentication between two parties A and B, in one of the following two ways:

- Prove the possession of a shared secret key (pre-shared or established), that only both parties know/share. This is typically done by generating/verifying a *MAC* (message authentication code) using the shared key. However, MACs do not provide non-repudiation service, since the used key is known to the two parties.
- Prove the possession of a private information (called private key) without revealing it, that the other party could verify using an authentic public information related to the prover (called public key). This is typically done by generating digital signatures using the signer's private key, while the receiver uses the signer's public key to check the signatures. Digital signatures provide non-repudiation service, *if and only if* the signer's private key is known to the signer only.

In [6], Fouda et al. investigate the authentication between two parties A and B of the SG: mainly a SM and a GW. Each entity possess a pair of self-generated private/public keys, while the public key is certified by one of the CAs (Certification Authorities) of the PKI. After verifying the certificate of each others (is valid and not revoked), A and B use the authenticated DH key establishment mechanism [4], in order to securely establish a secret shared key K_{A_B}, that both use to provide data source authentication. If non-repudiation is required, each party can use its private key to generate signatures.

In [8], So et al. propose an ID-based encryption and signature protocol for the AMI, based on IBC. The authors

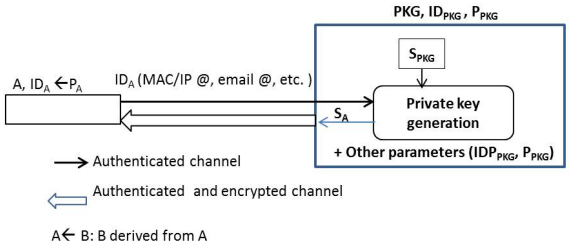

Figure 2. Private/Public Keys generation in IBC System

assume the existence of a widely trusted PKG that possesses a master private (S_{PKG})/public (P_{PKG}) keys. The PKG generates the ID-based private key of each entity A in the AMI using ID_A and S_{PKG}, while A's ID-based public key could be easily derived from A's identity ID_A (see Figure 2). Again, to provide data source authentication, A uses its private key to generate signatures over the messages it sends, while a receiver B uses ID_A to derive A's public key, then checks the authenticity of the messages received from A. Since the protocol is based on IBC, certificates are not used. The authenticity of A is implicitly checked if its signature is successfully verified, which means that A owns the right private key issued by the PKG.

In [10], Ayday et al. investigate authentication in end-customer's HAN, where each HAN contains: a GW, a SM, and a set of SAs. The authors assume that the utility plays the role of a widely trusted KDC in the SG, and that each GW possesses a unique IP address (serving as its ID) issued by an ISP (Internet Service Provider). The KDC shares a long-term unique secret key LTK_A with each entity A. Two entities A and B, with corresponding unique identifiers ID_A and ID_B, trying to establish a secure communication for the first time, send first an authenticated key-establishment request carrying both ID_A and ID_B, to the KDC. The KDC serves the requests as follows:

- The KDC checks that the messages are authentic using LTK_A and LTK_B.
- If the requests originate from a SM and a GW and are authentic, the KDC first uses the service of a third party providing localization information, to make sure that SM and GW are collocated (belong to the same HAN). Mainly, the KDC knows the location Loc_{SM} of SM due to the billing address of SM, then sends ID_{GW} and Loc_{SM} to the ISP (internet service provider) of GW, which could determine whether or not Loc_{GW}=Loc_{SM} . In this way, the KDC avoids a wrong/malicious key-establishment between a GW and SM of different HANs. If SM and GW are collocated, the KDC generates a secret key K_{SM_GW} and securely sends it encrypted using LTK_{SM} to SM and using LTK_{GW} to GW.
- If the key-establishment is between GW/SM and a SA,

 ISBN: 978-1-61208-189-2

the KDC first verifies that the GW and the SM related to the SA have already established a secret key. Finally, the KDC issues a secret key K_{SA_GW} (resp. K_{SA_SM}) and securely sends it to SA and GW (resp. SA and SM).

In [9], Yan et al. present an authentication and encryption protocol for the AMI, where SMs are interconnected through a multi-hop wireless network, and form a logical linear communication path to reach a remote collector node (GW). The authors assume the existence of trusted KDC (utility), which shares a secret key with each SM and securely issues a secret key to each pair of SMs in the communication path. A SM encrypts its reading using the key shared with the GW, and authenticates the data it sends (including data received from previous SMs in the path) using the key shared with the next SM in the path. In this way, each SM could verify the authenticity of the data it receives, and also guarantee the confidentiality of the metering data that it generates.

A. discussion

Authentication protocols based on symmetric-key cryptography coupled with KDC [9] [10], and those based on IBC coupled with a PKG [8], are known for their relatively low induced overheads (computation, storage, transmission), and lightweight management requirements (no need for PKI, since certificates are not used). In the other side, authentication protocols based on PKC are known for their relatively expensive overheads, especially for resources-constrained devices, and require the costly deployment of a PKI to issue and manage a large number of digital certificate. However, the fine performance of [8] [9] [10], comes at the expense of some security issues and some hard to fulfil assumptions:

- All the three protocols assume the existence of a *single trusted* entity (usually the utility) in the whole SG, playing the role of the PKG in [8] and the role of the KDC in [9] [10]. Assuming the existence of a single trusted entity (PKG or KDC) in the AMI is a hard to fulfil assumption, and do not scale to a so large network (millions of SMs, SAs, etc.). It is not trivial that the large number of manufacturers of SMs, SAs, GWs, e-cars, etc., would accept to trust the same single entity for key management. Moreover, would this single entity be able to efficiently manage the security of a large number of entities and systems involved in the AMI?
- In [8] based on IBC, the PKG issues the private keys of all entities of the SG. As a consequence, there is a key-escrow problem, and thus, non-repudiation service could not be guaranteed.
- In [9] [10], the KDC issues a secret shared key for any pair of communicating entities A and B in the AMI (SM, GW, SA, etc.). As a consequence, the privacy of communications in the AMI is not completely preserved, since the KDC can easily eavesdrop on the encrypted messages, or even impersonate any entity without being detected. Moreover, the end-customer's privacy in [10] is not preserved, since information related to the SAs in its HAN could be divulged during the key-establishment phase between a GW/SM and SAs.
- [10] assumes the existence of a GW per HAN/SM, whereas in practice there is one GW to serve a set of SMs. Requiring a GW per HAN, either means integrating a GW (with all its advanced features) with each SM, or deploying a separate GW per SM/HAN. Both solutions are financially expensive due to the large number of deployed SMs. Assuming that one GW serves a set of HANs, the use of ISP to check whether the SM and the GW are collocated will fail, since the GW and the SM are now in two different locations. As a consequence, preventing wrong/malicious key-establishment could not be guaranteed.
- Assuming that the utility plays the role of the KDC or the PKG is problematic, since in many countries (e.g., USA) several independent electricity utilities may operate in the same region. In this case, we will end up with several KDCs/PKGs, which make all the previous authentication protocols not directly applicable.

Finally, all the proposed protocols do not consider the case where SAs of HAN_i could successfully mutually authenticate with the SM of a neighboring HAN_j. This situation could make the SAs to be controlled by the SM of HAN_j instead of the SM of HAN_i. Meanwhile, the same problem occurs, except for [8], when establishing secure communications between a SM and the *right* GW: e.g., a SM communicate with the GW of a neighboring building and not the GW of the the local building.

III. Motivation, Assumptions and Notations

A. Motivation and Assumptions

To provide an efficient authentication for the AMI, we mainly make use of symmetric-key cryptography and IBC, and rarely consider using classical PKC (RSA, DSA, and even ECC) [4], since it requires the use of certificates, where certificate distribution/fetching and certificate validation will add extra overheads that some resource-constrained devices in the AMI (SMs, SAs, etc.) could not easily offer. The only exception is for PKGs, which still use classical ECC [11].

Moreover, in order to provide non-repudiation service and protect the end-customer's HAN privacy, we do not rely on a KDC for authentication and key management. Instead, we use a variant of IBC called certificate-less IBC [12], where each entity's private key is partially issued by the PKG, the other part of the key being securely issued by the entity itself. Using certificate-less IBC, we can now provide non-repudiation service for the AMI, which was not possible using the basic IBC as in [8].

Unlike [8] [9] [10], we assume the existence of several trusted key management authorities (PKG), where entities

 ISBN: 978-1-61208-189-2

Table I
USED NOTATIONS

SM	Smart meter
SA	Smart appliance
GW	Gateway
EP	Energy Provider
ID_A	unique identity of entity A
N_A	a nonce value generated by A
P_A	A's public-key
S_A	ID-based private key of A, or simply A's private key
PKG_i	the i^{th} Private Key Generator
P_{PKG_i}	Public key of PKG_i
S_{PKG_i}	master-secret Private key of PKG_i
MAC	Message Authentication Code
K_{A_B}	an l-bit secret shared key between entities A and B
MAC_K(M)	a MAC generated over message M using key K
Enc_K(M)	message M encrypted using key K
$\sigma_{S_A}(M)$	a signature generated over message M using S_A
p	a large prime of length $\mid p \mid$>160 bits
F_p	a finite field, F_p=[0, p-1]
$E(F_p)$	an elliptic curve defined over F_p
aP	scalar to point multiplication: addition of P a times
$x \parallel y$	concatenation of x and y

served by different PKGs could authenticate and establish secure communications. In this way, the scalability is improved, no single point of failure exists, and the end-customer's privacy is enhanced.

Finally, we assume that the SMs and the GWs deployed on the AMI network are owned and managed by the utility, which owns also the electricity distribution and/or transmission power network. In the other side, SAs inside the end-customer's HAN are owned and managed by the end-customer. In our paper, we make a distinction, which is not made by the other works, between the utility and the energy provider:

- The utility owns the physical electric infrastructure until the end-customer's electricity point of delivery (SM), over which electricity is delivered.
- The energy provider; with which the end-customer signs a contact, supplies the end-customer by energy (by buying it from energy produces), and bills it for the consumed energy. The utility is informed about each signed contract between any end-customer (SM) and any energy provider.

B. Notations

Table I summarizes the notations used through the remaining of the paper.

IV. OUR PROPOSED ID-BASED AUTHENTICATION PROTOCOL FOR THE AMI

In this section, we propose and describe an ID-based authentication protocol for the AMI, which provides source authentication, data integrity, non-repudiation and preserves the end-customer's HAN privacy.

Our protocol involves three phases: System setup phase, Node initialization/Private key generation phase and Data Source Authentication phase.

A. Phase I-System Setup

We assume the existence of t trusted entities in the SG, PKG_i $i = 1, ..., t$, playing the role of private key generation authorities. It is evident that t is infinitely negligible compared to the number of entities involved in the SG. The t PKGs agree on the use of the same elliptic curve $E(F_p)$, with the simplified domain parameters $Param_E = (a, b, p, n, P)$, where:

- p is the order of the finite field F_p over which is defined $E(F_p)$.
- $a, b \in F_p$ are the coefficients of $E(F_p)$.
- $P \in E(F_p)$ is a generator point of a cyclic subgroup of $E(F_p)$, and n is a big-prime and is the order of P.

The PKGs also agree on the use of two n-degree cyclic groups: $G_1 \subset E(F_p)$ (additive group, e.g., the subgroup of $E(F_p)$ defined by P) and G_T (multiplicative group, e.g., an extension field of F_n), on a symmetric pairing function e [13] and on two hash functions:
$H_1 : \{0,1\}^* \times G_1 \rightarrow G_1$ and $H_2 : G_T \times G_T \rightarrow \{0,1\}^l$.

e has the following properties [13]:

- *Bilinearity*: $e(aR, bS)$=$e(abR, S)$=$e(R, abS)$=$e(R, S)^{ab}$, $\forall$ a,$b \in F_n^*$, $\forall$ $R, S \in G_1$
- *Non-degeneracy*: $e(P, P) \neq 1_{G_T}$, where 1_{G_T} is the identity element of G_T, P is the generator of G_1.
- *Computability*: there exists an efficient algorithm implementing and computing e.

Then, each PKG_i performs the following:

- Picks a random master-secret private key $S_{PKG_i} \in {F_n}^*$ and computes its public-key P_{PKG_i}=$S_{PKG_i}P$, then sets its IBC system parameters: $Param_i$=$(n, G_1, G_T, e, P, P_{PKG_i}, H_1, H_2) \cup Param_E$.
- Securely gets the identity ID_{PKG_j} of each remaining PKG_j along with an authentic copy of P_{PKG_j}. Then, PKG_i issues a cross-domain certificate to each PKG_j, where T_{exp} is the certificate's expiration date:

$$Cert_{i \rightarrow j} = \underbrace{ID_{PKG_i}, ID_{PKG_j}, P_{PKG_j}, T_{exp}}_{\pi} \sigma_{S_{PKG_i}}(\pi)$$

  PKG_i makes $Cert_{i \rightarrow j}$ available by publishing it on a public repository referenced to it by $@Dir_i$.

For signature generation/verification over the certificates, we use the ECDSA signature scheme [11]. Figure 3, summarizes Phase I

B. Phase II-Node Initialization/Private Key Generation

Each entity involved in the AMI (SM, GW, SA, e-car, etc.) is issued a *partial* ID-based private key by one of the t PKGs at the manufacturing phase. Each manufacturer signs a contract with one or more PKGs, in order to securely provide each device it produces with its partial ID-based private key, along with the necessary IBC system parameters. We assume that the initialization step is performed over an

 ISBN: 978-1-61208-189-2

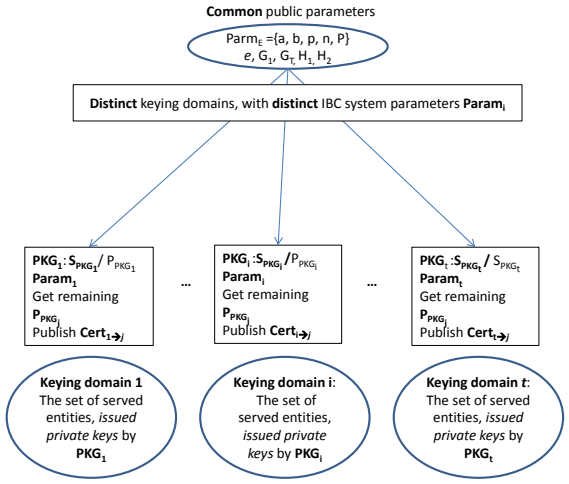

Figure 3. System Setup Phase

already established *secure channel* between PKG_i and the manufacturer.

Thus, a device A with a unique built-in identity ID_A (e.g., serial number, MAC address, IPv6 address, etc.), served by PKG_i will be *securely* initialized as follows:

- A is loaded with $Param_i$, $@Dir_i$
- Then, A picks a random $k_A \in F_n^*$ and sends to PKG_i: ID_A, k_AP.
- PKG_i sends to A its partial private key: D_A=$S_{PKG_i}H_1(ID_A, k_AP)$
- A computes its total ID-based private key S_A=k_AD_A, and sets P_A=$< k_AP$, $k_AP_{PKG_i} >$. A securely stores k_A and S_A, since it needs them for signature generation, public-key decryption and key-establishment.

At the end of the initialization, A is the *only* entity knowing S_A. Even PKG_i does not know it since it could not know k_A. A is said to belong to the keying domain i defined by PKG_i (see Figure 3). In the same way, each EP will securely get its partial private key, then computes its total private key.

Finally, each entity A=$\{SM, GW\}$ purshased by the utility, goes through a second step of initialization before its deployment. Mainly, the utility pre-loads the SM/GW with its identity $ID_{Utility}$ and a unique long term shared secret key LTK_A.

C. Phase III-Data Source Authentication

In this phase, two communicating parties A and B of the AMI (EP, GW, SM, SA, etc.), first mutually authenticate (the first time they met), then securely exchange messages. Two cases can be distinguished:

- **Case I**: Intra-domain communications: both A and B belong to the same keying domain i defined by PKG_i.
- **Case II**: Inter-domains communications: A and B belong to two different keying domains i and j, defined by PKG_i and PKG_j respectively.

We only consider **Case II**, since Case I is trivial. We consider three scenarios of communication, where all the communicating entities belong to two different keying domains:

- between EP and SM.
- between SM and GW.
- between SM and SA

For simplicity, we assume that during a communication the initiator always belongs to domain i, whereas the responder belongs to domain j. Also, we give the details of the ID-based signature generation/verification we use at the end of the section, and not for each scenario.

1) Communication between EP and SM: When the end-customer signs a new contract with an EP to be its energy supplier, the EP initiates a communication with the corresponding SM, by by setting a request $Req1$ (association with a new EP) and generating an ID-based signature over $M1$, then sends the following message to the SM:

$$\underbrace{Req1, ID_{EP}, P_{EP}, ID_{PKG_i}, N_{EP}, ID_{SM}}_{M1} \sigma_{S_{EP}}(M1) \quad (1)$$

Upon reception of (1), the SM checks the message freshness (not replayed) from the received nonce value N_{EP}. If expired or not already held, the SM checks from $@Dir_j$ whether or not there exists a certificate $Cert_{j\to i}$ issued from PKG_j to PKG_i. Assuming it exists, the SM checks $Cert_{j\to i}$'s validity using P_{PKG_j} following the ECDSA signature scheme [11]. If the certificate is valid (certificates signature is valid), the SM trusts P_{PKG_i}, and as consequence assumes that P_{EP} is a valid public-key. Finally, SM checks the signature on (1). If valid, the SM sends a request $Req2$ to the utility to check whether ID_{EP} is a valid energy provider, and whether or not it must accept $Req1$:

$$\underbrace{Req2, ID_{SM}, ID_{EP}, N_{SM}}_{M1a}, MAC_{LTK_{SM}}(M1a) \quad (2)$$

Upon reception of (2), the utility checks that the message is fresh and authentic, then checks if there is a new association between ID_{SM} and ID_{EP} (a new energy supply contract has been signed). If it is the case, the utility sends the following approval confirmation message (otherwise sends a deny message):

$$\underbrace{OK, ID_{EP}, ID_{SM}}_{M1b}, MAC_{LTK_{SM}}(M1b \parallel N_{SM}) \quad (3)$$

Upon reception of (3), the SM checks if it is fresh and authentic (MAC is valid) and that the response of is 'OK'.

 ISBN: 978-1-61208-189-2

If all verifications are positive, the SM considers that it has successfully authenticated the EP. The SM responds to the EP to both authenticate itself and also to confirm the acceptance of $Req1$:

$$\underbrace{OK, ID_{SM}, P_{SM}, ID_{PKG_j}, N_{SM}, ID_{EP},}_{M2}\ \sigma_{S_{SM}}(M2 \parallel N_{EP}) \quad (4)$$

Upon reception of (4), the EP verifies that it is fresh. If expired or not already held, the EP checks from $@Dir_i$ whether or not there exists a certificate $Cert_{i\to j}$. Assuming it exists and is valid, the EP trusts P_{PKG_j}, and consequently assumes that P_{SM} is valid. The EP checks the ID-based signature on 4. If the signature is valid and the response is 'OK', it considers that it has successfully authenticated the SM and sends the following message to conclude the mutual authentication phase:

$$Finish, \sigma_{S_{EP}}(Finish, ID_{EP}, ID_{SM}, N_{SM}) \quad (5)$$

Now, the EP and the SM can successfully generate and verify signatures over their exchanged messages, thus ensuring data source authentication and non-repudiation. They could also establish a shared secret key, as described in Section IV-C2, to secure their communications if non-repudiation is not mandatory.

2) Communication between SM and GW: A newly deployed SM in a residential building needs first to authenticate the GW of the building, before sending its metering data to the AMI head-end via this GW. The deployment of the SM is done by an authorized employee of the utility. To avoid associate the SM with a wrong GW (neighboring GW or a fake GW), all what is needed is that the employee indicates to the SM the ID_{GW} to which the SM needs to communicate. Assuming the employee securely initializes the SM with ID_{GW}, the following steps are performed between the SM and the GW:

The SM sets an *attach-req* request to ask to be attached to the GW, generates an ID-based signature over $M3$, then sends the following message to ID_{GW}

$$\underbrace{attach-req, ID_{SM}, P_{SM}, ID_{PKG_i}, N_{SM},}_{M3}\sigma_{S_{SM}}(M3) \quad (6)$$

Upon reception of (6), the GW checks its freshness. If expired or not already held, the GW checks from $@Dir_j$ whether or not there exists a certificate $Cert_{j\to i}$. Assuming it exists and is valid, the GW checks the validity of the signature on (6) using P_{SM}. If valid, using secret values k_{GW} and S_{GW}, and public-key P_{SM}, the GW computes a secret key K=K_{GW_SM}=

$$H_2(e(S_{GW}, k_{SM}P)e(k_{GW}H_1(ID_{SM}, k_{SM}P)k_{SM}P_{PKG_i}))$$

then sends the following message to the SM, to notify the acceptance of the request:

$$\underbrace{attach-ok, ID_{GW}, ID_{PKG_j}, N_{GW},}_{M4} MAC_K(M4 \parallel N_{SM}) \quad (7)$$

Upon reception of (7), the SM checks its freshness and makes sure that it originates from the pre-configured ID_{GW}. If expired or not already held, the SM checks from $@Dir_i$ the existence of a certificate $Cert_{i\to j}$. Assuming it exists and is valid, the SM computes using secret values k_{SM} and S_{SM}, and public-key P_{GW}, the secret key K=K_{SM_GW}=

$$H_2(e(S_{SM}, k_{GW}P)e(k_{SM}H_1(ID_{GW}, k_{GW}P), k_{GW}P_{PKG_j}))$$

then checks the received MAC. If the verification succeeds, then SM concludes that the GW is authentic and that K_{SM_GW}=K_{GW_SM} (otherwise detects that the GW is misbehaving and stops communication with it). Then, the SM sends the following message to authenticate itself to the GW:

$$finish, MAC_K(finish, ID_{SM}, ID_{GW}, N_{GW}) \quad (8)$$

The GW verifies the authenticity of (8). If the MAC is valid, the GW concludes that the SM is authentic and that K_{GW_SM}=K_{SM_GW}. Henceforth, the SM and the GW use K_{SM_GW} to provide data source authentication. However, if they need to provide non-repudiation, they can still use their private keys. Now let prove that K_{SM_GW}=K_{GW_SM}, for simplicity we omit H_2 in the proof.

We have K_{GW_SM}=$H_2(\alpha\ \beta)$, where

$$\begin{aligned}\alpha &= e(S_{GW}, k_{SM}P)) \\ &= e(k_{GW}S_{PKG_j}H_1(ID_{GW}, k_{GW}P), k_{SM}P) \\ &= e(k_{SM}H_1(ID_{GW}, k_{GW}P), k_{GW}S_{PKG_j}P) \\ &= e(k_{SM}H_1(ID_{GW}, k_{GW}P), k_{GW}P_{PKG_j})\end{aligned}$$

$$\begin{aligned}\beta &= e(k_{GW}H_1(ID_{SM}, k_{SM}P), k_{SM}P_{PKG_i}) \\ &= e(k_{GW}H_1(ID_{SM}, k_{SM}P), k_{SM}S_{PKG_i}P) \\ &= e(k_{SM}S_{PKG_i}H_1(ID_{SM}, k_{SM}P), k_{GW}P) \\ &= e(S_{SM}, k_{GW}P)\end{aligned}$$

From α and β, we can easily deduce that K_{GW_SM}=$H_2(\alpha\beta)$=$H_2(\beta\alpha)$=K_{SM_GW}

3) Communication between SM and SA: When a new SA is deployed in the end-customer's HAN, it first needs to authenticate itself to (associate itself with) the SM of the HAN, and also requires the authentication of the SM. To avoid associating a SA with a wrong SM (e.g., the SM of a neighboring HAN), the end-customer explicitly indicates to the SA ID_SM to which the SA needs to associate. Assume that the SA is provided with a data input interface (e.g., small keyboard), or could be connected to a PC and then be accessible through a software interface. In this case, the end-customer (SA's owner) could initialize the SA with the

 ISBN: 978-1-61208-189-2

appropriate ID_{SM} (the SM of its HAN). As a consequence, the SA and the SM could mutually authenticate and establish a secret key K_{SA_SM} for data source authentication, as described in Section IV-C2, without involving any online third party (KDC or PKG).

4) ID-based Signature Generation/Verification: We based our ID-based signature on the Hess ID-based signature scheme [14], while providing some modification to reflect the use of Certificate-less IBC. Let A belonging to domain i be the signer, B belonging to domain j be the verifier, and M the signed message. In addition, assume that B already trusts PKG_i. Moreover, S_A=$k_A S_{PKG_i} H_1(ID_A, k_A P)$ is A's private key and P_A=$< k_A P, k_A P_{PKG_i} >$ is A's public key.

A generates the signature $< R, v >$ as follows:

- Picks $k \in {F_n}^*$, and $P_1 \in {G_1}^*$, then computes

$$r = e(kP_1, P) \tag{9}$$

- Computes

$$v = H(M \parallel r) \tag{10}$$

and sets

$$R = vS_A + kP_1 \tag{11}$$

where H is a one-way hash function (e.g., SHA1)

- Outputs $\sigma_{S_A}(M)$=$< R, v >$

B verifies the signature $< R, v >$ using ID_A, P_A and P_{PKG_i} as follows:

- Computes r', where:

$$r' = e(R, P)e(-vH_1(ID_A, k_A P), k_A P_{PKG_i}) \tag{12}$$

- Accepts the signature *only and only if*

$$v = H(M \parallel r'). \tag{13}$$

Now, let prove that if the signature is verified (equation 13 held), then A really generated the signature over M using its private key S_A corresponding to ID_A and P_A, where S_A is partially generated PKG_i.

From (9), (10) and (13), we deduce that

$$r' = e(kP_1, P) \tag{14}$$

Finally, from (12) and (14), we have:

$$\begin{aligned} e(kP_1, P) &= e(R, P)e(-vH_1(ID_A, k_A P), k_A P_{PKG_i}) \\ &= e(R, P)e(-vH_1(ID_A, k_A P), k_A S_{PKG_i} P) \\ &= e(R, P)e(-vk_A S_{PKG_i}(ID_A, k_A P), P) \\ &= e(R, P)e(-vS_A, P) \\ &= e(R - vS_A, P) \end{aligned}$$

Now, we get that

$$e(kP_1, P) = e(R - vS_A, P)$$

By equality, term to term of the both pairing functions, we have

$$kP_1 = R - vS_A \Rightarrow R = vS_A + kP_1$$

Thus, we find here the initial signature as generated by A in 11.

V. Security Analysis and Comparison

Our ID-based authentication protocol for the AMI achieves secure authentication and non-repudiation. It also improves the privacy for the end-customer's. Any pair of communicating nodes of the AMI could mutually authenticate each other, then if authentic, securely exchange data either by signing them (if non-repudiation is required), or by establishing a shared secret key and generating MACs over the data (if non-repudiation is not required). Moreover, a SM and a SA could mutually authenticate without involving an online third party as the KDC in [9][10], thus, preserving the end-customer's privacy. Finally, the inclusion of a nonce value in each exchanged message protects the receiver from replay attacks. Indeed, a receiver accepts an authentic message from A as a *fresh*, *only and only if* the new nonce N_A carried in the message is grater than the last nonce received from A.

In our proposed solution, two nodes A and B are able to mutually authenticate, *if and only if*:

- They belong to the same keying domain i: In this case, both are issued a partial private key from the trusted PKG PKG_i.
- They belong to two different keying domains: In this case, they could not directly trust each other's public-key, since they do not trust the public-key of the PKG of the other domain. Each node needs to get a cross-domain certificate, issued by the PKG of its local domain to certify the public key of the PKG of the other domain. If such cross-certificate exists, then both entities could mutually authenticate, and also establish a shared key to protect their communications. If such certificate does not exist, then they will not be able to communicate.

A node X cheating on its ID or the ID of its PKG, is not able to pass the authentication phase in the three scenarios, since it could not have the appropriate private key corresponding to its identity and to its public-key. Consequently, X is not able to generate a valid signature that the other side will successfully verify, and cannot generate the same shared key as generated by the other party. As a consequence, X will be detected as misbehaving/cheating, and the communication with him will be stopped.

Our ID-based authentication protocol covers communications between the SM and the EP, a feature that is not described in the related works presented in Section II. Authentication in our protocol is very useful, especially since the end-customer is able to freely move from one EP

 ISBN: 978-1-61208-189-2

to another, in the context of SG. Also, in the new SG, the tendency is to make a separation between the utility (infrastructure provider) and the EP (electricity provider), since they are two separate entities. However, the intervention of the utility will remain, as seen in Section IV-C1. Indeed, the utility protects the SM from malicious/illegal association to any new EP.

The use of certificate-less IBC removes the key-escrow problem in [8], and thus guarantees non-repudiation. Indeed, an entity A is the only entity knowing its private key S_A. Thus, A could not repute a signature that it has generated and that could be verified using P_A, by claiming that another party (e.g., PKG_i) generated it, since PKG_i does not know S_A.

Finally, for authentication between SM/GW, and SM/SA, an authorized human intervention is performed to avoid a wrong/malicious association. Indeed, for SM/GW, an authorized field personal from the utility will indicate to the SM the identity of the associated GW, in order to send its meter readings and receive messages from utility and its EP. For SM/SA, the owner of the SA will indicate the identity of the SM to which the SA need to communicate with.

VI. Conclusion and Perspectives

Authentication is an important requirement to protect the AMI from several attacks, such as impersonation and data modification. In this paper, we present an ID-based authentication protocol for the AMI that induces low overheads, provides non-repudiation and authentication services, allows efficient key-establishment and preserves the end-customers privacy. Moreover, our solution is scalable since it considers several key management authorities. As a future work, we will evaluate the performances of the protocol, through simulation and implementation. Then, we will extend it to provide also confidentiality for communication in AMI, and enforce end-customers privacy through anonymization techniques, so that the generated metering data could not be linked to a particular SM/end-customer.

References

[1] D. P. Chassin, "What can the smart grid do for you? and what can you do for the smart grid," *Electricity Journal, Elsevier*, vol. 23, no. 5, pp. 57–63, June 2010.

[2] ASAP-SG, "The advanced security acceleration project," Open SG User Group, Online, http://osgug.ucaiug.org/utilisec/amisec/, (retrieved: January 2012), Technical Guidline, June 2010.

[3] F. M. Cleveland, "Cyber security issues for advanced metering infrasttructure," in *IEEE Power and Energy Society General Meeting-Conversion and Delivery of Electrical Energy in the* 21^{st} *Century*. Pittsburgh, PA: IEEE, July 2008, pp. 1–5.

[4] A. J. Menzes, P. C. V. Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*. CRC Press, 1997.

[5] C. Farell and S. Adams, "Internet x.509 public key infrastructure certificate management ptotocols. rfc 2510."

[6] M. M. Fouda, Z. M. Dadlulah, N. Kato, R. Lu, and X. Shen, "Towards a light-weight message authentication mechanism tailored for smart grid communications," in *Int. Workshop on Security in Computers, Networking and Commmunications*. Shanghai, China: IEEE, April 2011, pp. 1035–1040.

[7] L. Chen, "Identity-based cryptography," International School on Foundations of Security Analyses and Design, http://www.sti.uniurb.it/events/fosad06/ (retrieved: January 2012), 2006.

[8] H. K. H. So, S. H. M. Kwok, E. Y. Lam, and L. King-Shan, "Zero-configuration identity-based signcryption scheme for smart grid," in *IEEE Int. Conf. on Smart Grid Communications*, Gaithersburg, USA, October 2010, pp. 321–326.

[9] Y. Yan, Y. Qian, and H. Sharif, "A secure and reliable in-network collaborative communication scheme for advanced metering infrastructure in smart grid," in *Wireless Communications and Networking Conference*. Caucun, Mexique: IEEE, March 2011, pp. 909–914.

[10] E. Ayday and S. Rajagopal, "Secure, intuitive and low-cost device authentication for smart grid networks," in *IEEE Consumer Communications Networking Conference*, Las Vegas, USA, January 2011, pp. 1161–1165.

[11] D. Hankerson, A. J. Menzes, and S. Vanstone, *Guide to Elliptic Curve Cryptography*. New-York: Springer-Verlag, 2004.

[12] G. Omahen, A. Souvent, and B. Luskovec, "Advanced metering infrastructure for slovenia," in *IEEE Conf. on Electricity Distribution*, Prague, Slovenia, June 2009, pp. 673–676.

[13] A. Menzes, "Introduction to pairing-based cryptography," Online, http://www.math.uwaterloo.ca/ ajmeneze/publications/pairings.pdf (retrieved: January 2012).

[14] F. Hess, "Efficient identity based signature schemes based on pairings," in *SAC02 Revised Papers from the* 9^{th} *Annual International Workshop on Selected Areas in Cryptography*. London UK: Springer-Verlag, 2003, pp. 310–324.

 ISBN: 978-1-61208-189-2