A Chain Based Signature Scheme for Uplink and Downlink Communications in AMI Networks

Samer Khasawneh(✉) and Michel Kadoch

Department of Electrical Engineering, École de Technologie Supérieure,
University of Quebec, 1100 Rue Notre-Dame O, Montreal, Canada
samer.khasawneh.1@ens.etsmtl.ca,
michel.kadoch@etsmtl.ca

Abstract. Smart grid is an electric infrastructure that makes extensive use of communication and information technology making it a surface for numerous cyber-security threats. In this research, we propose an authentication scheme for downlink and uplink communications in the advanced metering infrastructure network. The proposal is based on chain based signature with some modifications to tackle its computation and storage overhead. Besides, the proposal integrates symmetric encryption with the signature scheme to ensure data privacy and confidentiality. Our analysis proves that the proposed scheme is resilient against numerous known attacks and is efficient in terms of computation cost and ciphertext size.

Keywords: Smart grid · AMI network · Security · Chain based signature

1 Introduction

In the past few years, strong pressure is generated to switch from the power generation that mostly based on fossil sources towards a modernistic smart system that highly incorporates renewable forms of energy [1]. The pressure was derived by the strong growth in electricity demand in addition to the emerging large quantities of distributed renewable energy sources.

The main attribute that characterizes the new power grid is integrating modern communication and information technology into the grid, making it smarter. The recent advances in communication and information technology can optimize the power grid performance by enabling us to generate, monitor, collect, analyze and react to data describing the grid's physical condition. However, it is no surprise that integrating communication and information technologies will result in a complex system-of-systems which requires a sophisticated architecture that is inherently Quality-of-Service (QoS) aware. The most widely accepted smart grid architecture viewpoint is the one that comprises seven domains: market, operations, service provides, bulk generation, transmission, distribution and customer [2], where the later four domains are the classic power system components.

A.-S. K. Pathan et al. (Eds.): SGIoT 2018, LNICST 256, pp. 85–99, 2019.
https://doi.org/10.1007/978-3-030-05928-6_9

Advanced Metering Infrastructure (AMI) is the architecture that comprises smart meters at the customer's premises, data concentrators (gateways) and a supervisory node that acts as the AMI headend. Smart meters have multiple communication interfaces and are connected to various devices through a Home Area Network (HAN). It can collect information from the connected smart appliances to facilitate real-time billing. Smart meters can also issue commands to enforce peak demand management. Data concentrators preprocess the data received from the smart meters before having the data transmitted to AMI headend. Concentrators are stationed in physically secure locations such as substations. The supervisory headend node is located at the utility, within the company network. Basically, it acts on the smart meter's data and can issue several control commands such as pricing information updates, remote load control and demand response project's announcements. AMI systems enable near real-time pricing information and load exchange between the smart meters and utility business systems [3].

From the aforementioned description, it could be noted that AMI network realizes computerized two way communication between the metering network devices (in opposite to the conventional power grid that implements one way communication). Two-way communication is a smart grid feature that promotes implementing new functionalities such as Demand-Response, load shedding, peak shaving and self-healing [4]. In this case, QoS, reliability and real-time communication are critical performance factors.

1.1 Smart Grid Security

Smart grid is expected to optimize energy management, integrate renewable energy sources and introduce efficient billing schemes. Attaining such functionalities requires extensive use of information and communication technologies on a large-scale landscape. Accordingly, smart grid will be subject to significant cyber-security threats that will have negative impact on the grid services. Examples of such possible attacks are: Denial of Service (DoS), spoofing, replay, impersonation, data injection and privacy exposure (invasion) attacks. The degree of the damage caused by a cyber-attack depends mainly on the attacker skills and resources.

Achieving secure smart grid communication is crucial yet a challenging task for a number of reasons. First of all, the vast majority of the smart grid devices (especially the AMI devices) are equipped with limited storage, processing and communication capabilities. For this reason, some data encryption and authentication schemes could not be adapted for the smart grid. In addition, the lifetime of the power hardware is expected to be much longer than the information technology solutions. Therefore, a perfectly secure communication scheme is not expected to function during the whole lifetime of the power hardware. Another issue is the smart grid openness. The smart grid spans very large geographical areas and utilizes power devices from different manufactures which requires extremely high degree of interoperability between the grid systems and components. Finally, applying security measures may have counterproductive impact on the smart grid goals. For instant, a time critical packet may miss its deadline with the advanced authentication and integrity checks in place.

1.2 Our Contribution

The following are our contributions in this paper:

- We classified the AMI traffic into downlink and uplink traffic and associated the good transmission mode(s) for each one of them. This enables the network devices to efficiently generate and share the cryptographic keys without the need to maintain unnecessary keys.
- We have modified the basic chain based signature model to improve its computation and storage overhead. The modified signature scheme is used to propose an authentication model for downlink and uplink communications in AMI networks. A symmetric encryption is integrated with the signature scheme to ensure data confidentiality.
- Security analysis and performance evaluation are carried out to assess the feasibility of the proposed scheme. The results demonstrate that the proposed scheme is efficient in terms of computation cost and ciphertext size. In addition, it is capable to withstand various security attacks.

The rest of the paper is organized as follow. Section 2 reviews the related work. Section 3 demonstrates the models, design goals and background. The proposed encryption and signature scheme is illustrated in Sect. 4. The performance of the proposed scheme is presented in Sect. 4. Security analysis is presented in Sect. 5. Finally, the paper is concluded in Sect. 6.

2 Related Work

Nowadays, smart grid security is considered one of the most active research areas that attracted the researcher's attention. Despite the fact that the problem of security in the smart grid has not been fully identified, several researches have been proposed in the literature to address it. A zero-configuration identity-based signcryption for end-to-end communication in the advanced Metering Infrastructure (AMI) networks is proposed in [5]. The proposal has two phases of operation: registration phase and data transmission phase. In the registration phase, a device communicates with a Key Generation Server (KGS) to obtain a private key. The private key is used either to decrypt a received message or to sign a message before transmitting it. In the transmission phase, the sender calculates the receiver's public key using information derived from the receiver's identity and encrypts the message using the public key calculated. As the public keys are generated from information that is derived from the sender identity, the scheme achieves low computation overhead.

Anonymous Key Distribution (AKD) scheme for smart grid networks is proposed in [6]. The scheme is based on identity based elliptic curve cryptography to provide smart meter anonymity and mutual authentication. The scheme has several advantages such as: avoiding the need for third trusted party and achieving low computation and communication overhead compared to other schemes [7]. The proposal is resilient against data and impersonation based attacks.

Saxena et al. proposed a signature scheme for delivering authentic critical and non-critical commands in smart grid networks [8]. The proposed scheme is based on a set of cryptographic hashing functions to generate the message hash code. The code is splitted into several substrings with a predetermined length. The hashing functions are also used to generate the asymmetric keys (public/private) that will be used for signing the messages. Despite the fact that the scheme is secure against some authentications attacks, it has one major limitation. The authors assume that the signature is only constructed at the supervisory node; thereby alternative nodes such as smart meters don't sign their messages. Consequently, customer privacy could not be efficiently preserved.

An identity based signcryption technique for smart grid residential tree network is presented in [9]. The model employs bilinear pairing signcryption and destination concealing to achieve data integrity, authenticity and to preserve customer privacy. The proposed technique is designed to secure downlink communication between the control center and smart meters. The control center simultaneously encrypts and signs the messages before forwarding them to the smart meters. The authors show that the proposed technique is efficient in terms of computation cost and ciphertext length when compared to other schemes such as [10, 11]. However, the functionality of the proposed scheme is considered limited as the security measures are applied to messages generated by the control center. The uplink traffic generated by smart meters is not secured although it usually carries privacy-sensitive information. Further, the model supports unicast transmission mode only.

Mahmood et al. proposed a mutual authentication protocol for smart grid devices in [12]. The protocol utilizes elliptic curve cryptography and hashing functions to achieve data authenticity. The authentication protocol depends on the Elliptic Curve Discrete Logarithmic Problem (ECDLP) to attain prefect forward secrecy. Further, the scheme is constructed to withstand different attacks such as replay, impersonation and Man-in-the-middle attacks. The authors declare that the proposed authentication procedure is lightweighted in terms of computation complexity in addition to communication and memory overhead. The performance of the proposed scheme in one-to-many communication paradigms is suspected.

3 Models, Design Goals and Background

3.1 Network and Communication Model

In our scheme, we assume the AMI network that comprises three devices namely: smart meters (SMs), gateways (GWs) and Supervisory Control Center (SCC) as shown in Fig. 1. SMs are responsible for reporting energy consumption in addition to receiving billing information, thereby are equipped with limited computation and communication power. SCC has unlimited computation and communication power enabling it to manage the grid operation through performing critical tasks such as load shedding and demand response handling. Gateways have important rule in routing information in bidirectional paths from/to the SCC. The three devices can simultaneously and asynchronously perform signcryption operations at any time.

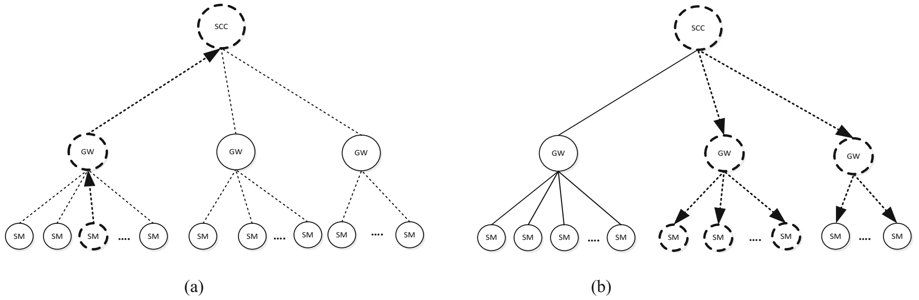


Fig. 1. The AMI communication models assumed in this paper. (a) Unicast uplink communication. (b) Multicast downlink communication

We have examined the communication paradigms that may exist between the three AMI devices and we have found that a communication in AMI network can fall in one of two categories:

a) *Uplink communication*

This communication is carried out when a lower layer device D_{LL} sends message to a higher layer device D_{HL}. For example, a smart meter transmitting to the corresponding gateway or a gateway that routes message to SSC. The transmission mode for such communication is unicast only.

- Unicast $(GW_i \rightarrow SSC, SM_i \rightarrow GW_i)$

Figure 1(a) shows an example of this case when a smart meter is reporting the customer's energy consumption to the corresponding gateway.

b) *Downlink communication*

In this case, a higher layer device D_{HL} sends message to a lower layer device D_{LL}. It could be the SCC transmitting to the corresponding gateway(s) or a gateway is routing messages to the corresponding smart meters(s). The transmission mode for such communication is unicast, multicast or broadcast.

- Unicast $(SSC \rightarrow GW_i, GW_i \rightarrow SM_i)$
  Remote load control is an example of this communication scheme. The supervisory node continuously monitors customer's consumption and can issue special command to enforce peak management.
- Multicast $(SSC \rightarrow GW_i, GW_i \rightarrow SM_i) i \in \{0, 1, \ldots l\} l < L$
  An example of downlink multicast AMI communication is shown in Fig. 1(b). Price updating and remote load control commands are triggered by the supervisory node and disseminated to certain Demand-Response projects in multicast transmission mode.
- Broadcast $(SSC \rightarrow GW_i, GW_i \rightarrow SM_i) i \in \{0, 1, \ldots L\}$

This case is similar to the previous one, except having the AMI headend (supervisory node) or gateway transmits the command to all AMI devices. Publishing of DR projects is an example of this communication style.

3.2 Adversarial Model

In our threat model, we assume an external polynomial time adversary $\mathcal{A}$ with a sufficient knowledge and computation power. The adversary $\mathcal{A}$ can access the public communication channel and capture network messages. Accordingly, he can eavesdrop, analyze, inject, replay, modify and delete data from the communication channel. Additionally $\mathcal{A}$ can compromise any smart meter (SM_i) or gateway (GW_i) and launch identity theft attacks later on. We assume the supervisory node (SCC) is securely suited within the utility premises and could not be compromised by the attacker.

3.3 Design Goals

Given the aforementioned adversarial model, our goal is to design an efficient encryption and authentication scheme for downlink and uplink communications in AMI networks. The model will be designed to take into consideration the requirements of each transmission mode for every communication direction. Practically, we aim to achieve the following three goals:

- *Authentication and Integrity.* The proposed scheme must guarantee that AMI messages injected or modified by the adversary $\mathcal{A}$ do not go undetected.
- *Confidentiality and privacy preservation.* Network messages (especially metering data) could be disclosed to authorize AMI participants only. The proposal should ensure that customer privacy never been infringed.
- *Efficiency.* The proposed encryption and authentication scheme should be lightweighted. It should be competent in terms of computation and communication cost compared to existing schemes.

3.4 Background

Chain based signature

t-time signature schemes could be constructed by combining a tuple of t independently generated private keys to form the private key, where the public key is constructed similarly. Each private/public key is used for a single signature generation/verification. Consequently, t signatures are generated using the tuple of private keys. The upper bound t should be determined in advance during the keys generation process. Such signature scheme has two main limitations. First, the number of signatures that could be constructed before re-invoking the key generation function is bounded. Second, the size of the cryptographic keys is large as each key consists of t individually generated keys. Chain-based signature scheme can achieve better performance in terms of key generation by allowing the signer to generate the cryptographic keys on the fly as needed.

Assume Ĉ = (***Gen***, ***Sign***, ***Vrfy***) a chain-based signature scheme, where ***Gen*** is the random key generation function that is used to generate the public and private keys on demand, ***Sign*** is the one way function that is used to construct the digital signature (σ) and ***Vrfy*** is the signature verification function. The operation of the chain-based scheme starts by having the signer generate a pair of cryptographic keys PK_0 and SK_0. In order to sign the first message m_0, the signer generates additional pair of keys (PK_1, SK_1), append the public key PK_1 to the message m_0 and signed the result using ***Sign*** and the private key SK_0 to obtain the signature $\sigma_0 \leftarrow Sign_{sk_0}(m_0||PK_1)$. PK_1 is generated and shared with the verifier in advance to enable verifying the message that will be signed next. Additionally, the signer has to store the state $\{m_0, PK_1, SK_1, \sigma_0\ \}$ to enable obtaining correct chaining between the signed messages. Subsequent messages are signed using the same procedure. For example, to sign the i^{th} message m_i, ***Gen*** is invoked to generate the key pair (PK_{i+1}, SK_{i+1}), m_i and PK_{i+1} are signed using SK_i to obtain the signature $\sigma_i \leftarrow Sign_{sk_i}(m_i||PK_{i+1})$. The state $\{m_i, PK_{i+1}, SK_{i+1}, \sigma_i\}_{j=0}^{i-1}$ is added to the signer states. The signature that will be outputted includes σ_i, the next public key in the chain (PK_{i+1}) and the states $\left\{m_i, PK_{i+1}, SK_{i+1}, \hat{S}_i\right\}_{j=0}^{i-1}$ as well.

Verifying the signature σ_i of message m_i requires the verifier to validate (a) the lastly generated public key PK_{i+1} that is attached to m_i (b) the link between every consecutive public keys PK_j and PK_{j+1} in the signature chain. The verification function outputs 1 (as an indication of successful verification) if and only if ***Vrfy*** $(PK_j, \hat{S}, m_j||PK_{j+1})$ outputs 1 for all $j \in \{0, \ldots, i-1\}$. Accordingly, the verification process begins with the firstly generated public key PK_0 and goes with all public keys on the chain until PK_{i-1}.

Elliptic Curve Digital Signature Algorithm (ECDSA)
ECDSA is a public key algorithm that was accepted in 1999 as an ANSI standard as a substitute to the Digital Signature Algorithm (DSA). It is based on elliptic curve cryptography; which yields a security level compared to that of other public key schemes but with smaller key length. The strength of ECDSA comes from the need for solving the Elliptic Curve Discrete Logarithm Problem (ECDLP). ECDSA involves the use of three algorithms: key generation, signing construction, and signature verification. The key generation algorithm computes the private key (d) and the public key $(Q = dG)$ to use in the verification and signature, respectively. In the proposed scheme, we implement the chain based signature using ECDSA.

4 The Proposed Scheme

In this section, we propose a crossbred encryption and signature scheme to confront confidentiality, integrity and authentication threats in AMI network. Symmetric encryption is employed to suit the requirement of low computation overhead.

In our scheme, the digital signature is constructed using a low complexity chain based algorithm. We assume a two-way AMI communication network where meters and SCC bidirectionally exchange data and control messages throughout the intermediate gateways. We address two communications flows namely: downlink and uplink. Downlink

Table 1. Notation guide

Notation	Description	Notation	Description
a, b, q, G, n, h	Elliptic curve parameters	$\boldsymbol{Sign}_{KS_i}$	Signature generation function
$\aleph_0$	An adversary	$\boldsymbol{Vrfy}_{PK_i}$	Signature verification function
D_{LL}	Lower layer device	M_t	AMI message
D_{HL}	Higher layer device	$\overline{M}_t$	Encrypted AMI message
$\aleph_t$	Initial nonce	ID_{LL}	ID of Lower Layer device
$\aleph_t$	Nonce of session t	σ	Digital signature
$h_1(.), h_2(.)$	Hashing functions	l	Multicast domain size
k_{sh}	DH shared symmetric key	L	Broadcast domain size
k_U, k_M, k_B	Symmetric encryption keys for unicast, multicast, broadcast	$\lvert point \rvert$	Size of elliptic curve point including x and y coordinates
$KP_{UD}, KP_{MD}, KP_{BD}$	Public downlink unicast, multicast, broadcast keys for signature verification	$PGen(b)^1$	Asymmetric key generation function
$KS_{UD}, KS_{MD}, KS_{BD}$	Private downlink unicast, multicast, broadcast keys for signature generation	$SGen(b)^1$	Symmetric key generation function
PK_{UU}	Public uplink unicast signature verification key	$SYMM.ENC_k$	Symmetric encryption algorithm
SK_{UU}	Private uplink unicast, multicast, broadcast keys for signature verification	$SYMM.DEC_k$	Symmetric decryption algorithm

traffic is disseminated by SCC towards the smart meters and could have unicast, multicast or broadcast modes, whereas uplink traffic originated from the meters is unicast. The proposed scheme runs in two phases that are described in the following subsections.

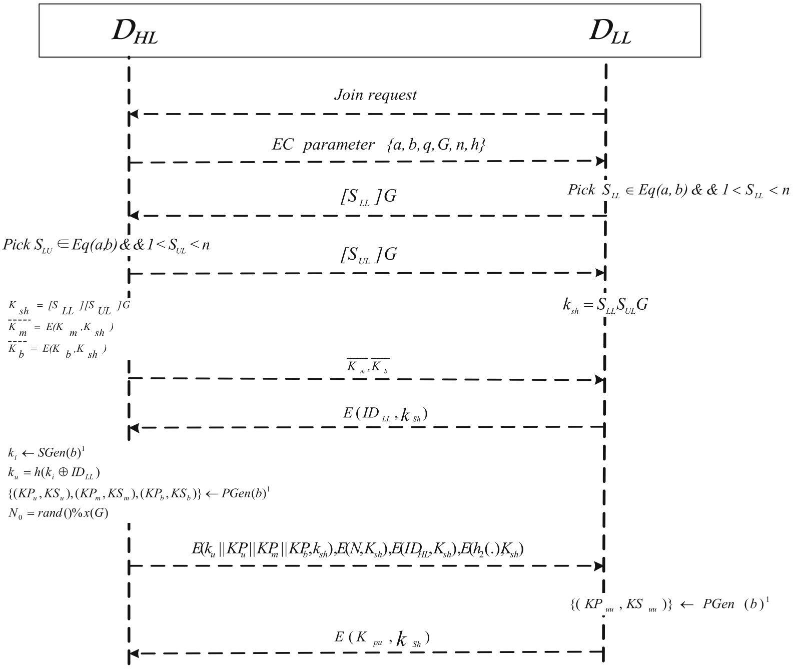
4.1 Initialization Phase

The phase is demanded when a new smart meter or gateway joins the AMI network. The device initiates the initialization procedure with the corresponding gateway or SCC, respectively. In our scheme, the device that initiates the procedure is the *Lower Layer Device* D_{LL}, while the device that receives the initialization request is the *Higher Layer Device* D_{HL}. Consequently, D_{LL} is a smart meter or gateway and D_{HL} is a gateway or the SCC. This phase is required to enable D_{LL} and D_{HL} to securely set up the cryptographic parameters (keys and hash functions) over the inherently insecure AMI channels. D_{HL} assembles the elliptic curve parameters and shares them with D_{LL} to enabling generating the shared secret key K_{sh} based on Elliptic Curve Diffie Hellman (ECDH) protocol.

Downlink traffic is generated by D_{HL} and is transmitted to one or more D_{LL} devices in unicast, multicast and broadcast mode. Therefore, three secret keys $\{k_U, k_M, k_B\}$ are shared with D_{LL} to enable decrypting D_{HL} message's. Similarly, three public keys $\{KP_{UD}, KP_{MD}, KP_{BD}\}$ are securely shared with D_{LL} to enable verifying signatures constructed using the private keys $\{KS_{UD}, KS_{MD}, KS_{BD}\}$. On the other hand, uplink traffic is generated by D_{LL} and is transmitted to a single D_{HL} in unicast mode only. Consequently, one public key PK_{UU} need to be shared with D_{HL} to enable it verifies the signatures constructed using D_{LL} private key SK_{UU}. Additionally, D_{HL} randomly chooses an initial nonce value N_0 and shares it with D_{LL}. As we will demonstrate later in Sect. 6, using nonce can detect replay attack.

4.2 Encryption and Authentication Phase

In our model, two or more AMI participants can communicate securely by exchanging encrypted and signed messages. Symmetric cryptography is used to encrypt and decrypt the message content, while chain based signature is used to construct and verify the message signature (Fig. 2 and Table 1).



D_{HL}
D_{LL}
Join request
EC parameter $\{a, b, q, G, n, h\}$
$[S_{LL}]G$
Pick $S_{LL} \in Eq(a, b)$ & & $1 < S_{LL} < n$
Pick $S_{LU} \in Eq(a,b)$ & & $1 < S_{UL} < n$
$[S_{UL}]G$
$K_{sh} = [S_{LL}][S_{UL}]G$
$\overline{K_m} = E(K_m, K_{sh})$
$\overline{K_b} = E(K_b, K_{sh})$
$k_{sh} = S_{LL} S_{UL} G$
$\overline{K_m}, \overline{K_b}$
$E(ID_{LL}, k_{Sh})$
$k_i \leftarrow SGen(b)^1$
$k_u = h(k_i \oplus ID_{LL})$
$\{(KP_u, KS_u), (KP_m, KS_m), (KP_b, KS_b)\} \leftarrow PGen(b)^1$
$N_0 = rand()\% x(G)$
$E(k_u || KP_u || KP_m || KP_b, k_{sh}), E(N, K_{sh}), E(ID_{HL}, K_{sh}), E(h_2(.) K_{sh})$
$\{(KP_{uu}, KS_{uu})\} \leftarrow PGen(b)^1$
$E(K_{pu}, k_{Sh})$

Fig. 2. The initialization phase

Now, suppose the multicast domain $\{D_{HL_1} \rightarrow D_{LL_i}, D_{LL_j} \ldots D_{LL_n}\}$ to enable gateway GW_1 delivering a remote load control message securely to the set of connect smart meters $\{SM_i, SM_j, \ldots \ldots SM_n\}$. The communication shall proceed as follows:

Step 1: Message construction Encryption $(D_{HL_1} : \{\overline{M}_t\})$

In the proposed model, the message to be encrypted has three components: the data content D, the nonce value $\mathbb{N}_t = \mathbb{N}_{t-1} + 1$ and a public key PK (multicast public key $PK_{MD_{t+1}}$ in this case). The nonce value is used to keep the parties in sync to ensure withstanding replay attack. Therefore, the transmitted nonce value is, $\mathbb{N}_{t-1}$ where $SYMM.ENC_{k_{MD}}(D||\mathbb{N}_t||PK_{MD_{t+1}})$ is the nonce used in the previous session. In order to implement the chain based signature scheme, the public key that will be used by the receiver to verify the next signature must be sent a priori. Therefore, Asymmetric key generation function $PGEN$ is used to generate a pair of keys (PK_{t+1}, SK_{t+1}). The public key $PK_{MD_{t+1}}$ is attached to t^{th} message. The encrypted message $\overline{M}_t$ is $D_{LL_i}, D_{LL_j} \ldots D_{LL_n} : \{D, \mathbb{N}_t, PK_{MD_{t+1}}\}$.

Step 2: Signature construction $(D_{HL_1} : \{\sigma_t\})$

As discussed in Sect. 3.4, the chain based signature outputs the state $\{m_i, PK_{i+1}, SK_{i+1}, \sigma_i\}_{j=0}^{i-1}$ with the signature σ_i to enable the receiver verifying the signature. Maintaining and processing such state leads to considerable processing and storage overhead. In the proposed model, we included the public key that will be used to verify the signature with each message and in the initialization phase as well. This, in addition to the fact the each AMI device is always communicating with the same device(s) eliminate the need for outputting such state with each signature. Accordingly, the signature σ_t is constructed using $SK_{MD_t}, h_2(.)$ and the one way signature generation function ***Sign*** as $\sigma_t = \boldsymbol{sign}_{SK_{MD_t}}\left(h_2\left(\overline{M}_t\right)\right)$.

Step 3: Multicast Transmission $(D_{HL_1} \rightarrow D_{LL_i}, D_{LL_j} \ldots D_{LL_n} : \{M_t, \sigma_t\})$

D_{HL_1} transmits the encrypted message $\overline{M}_t$ along with the signature σ_t for each $D_{LL} \in \{D_{LL_i}, D_{LL_j} \ldots D_{LL_n}\}$

Step 4: Signature verification $(D_{LL_i}, D_{LL_j} \ldots D_{LL_n} : \boldsymbol{Vrfy}_{Pk}(.))$

Every D_{LL} in the multicast domain $\{D_{LL_i}, D_{LL_j} \ldots D_{LL_n}\}$ that receives the signature will use the multicast public key PK_{MD_t}, the hashing function $h_2(.)$ and the one way signature verification function ***Vrfy*** to verify the signature. The signature is accepted if and only if $\boldsymbol{Vrfy}_{PK_{MD_t}}\left(h_2\left(\overline{M}_t\right), \sigma_t\right) = 1$, otherwise the signature is rejected and impersonation attack is reported.

Step 5: Message decryption

If the signature σ_t is accepted, the multicast domain members $\{D_{LL_i}, D_{LL_j} \ldots D_{LL_n}\}$ individually decrypts $\overline{M}_t$ using the multicast downlink key k_{MD} to obtain $M_t = SYMM.DEC_{k_{MD}}\left(\overline{M}_t\right)$. Then, the received nonce value is checked to detect if the message is replayed. Replay attack is detected if and in this case the message is ignored and replay attack is reported. Otherwise, the data content D is processed and the received public key $PK_{MD_{t+1}}$ is stored to enable verifying the message that will be received next. is updated with as well.

Performance Evaluation

The efficiency of the proposed encryption and signature scheme can be evaluated in terms of the computation cost and ciphertext length. In this section, we present the performance of the proposed scheme and compare it with the signcryption model presented in [9].

4.3 Computation Cost

According to our scheme, the computation cost is the time overhead required to encrypt-sign the plaintext or verify-decrypt the ciphertext message. We implemented the chain based signature scheme using ECDSA where the elliptic curve point multiplication represents the most computation intensive operation. Symmetric cryptography, on the other hand, is very fast compared to public key cryptography. Therefore, the computation cost of our scheme is determined mainly by the time required to construct (sign) or verify a signature using the proposed chain based scheme, where point multiplication dominates ECDSA time.

Table 2 demonstrates a comparison between the computation cost required by our scheme and the signcryption scheme presented in [9]. The time required to generate the ciphertext in our mode is $T_{symm} + T_{mul}$ compared to $4 \times T_{mul} + T_{pair}$ for the signcryption scheme. Moreover, the time needed to recover the plaintext in our scheme is $2 \times T_{mul} + T_{symm}$ compared to $T_{mul} + 4 \times T_{pair}$ for the signcryption scheme.

Table 2. Computation cost: the proposed model vs. the signcryption scheme [9]

Scheme	Symmetric cryptography		EC point multiplication (T_{mul})		Pairing computation (T_{pair})	
	Our model	Model in [9]	Our model	Model in [9]	Our model	Model in [9]
Cipher-text generation	T_{symm}	–	1	4	–	1
Plaint-text recovering	T_{symm}	–	2	1	–	4

In order to show the numerical computation cost, we have done a computer simulation for two AMI networks, one implements our scheme and the another implements the signcryption scheme proposed in [9]. The simulation was executed on an Intel Pentium IV 3.1-GHz machine with 8 GB RAM. We have chosen AES-128 as the symmetric cryptography algorithm and SPEC112r1 standard [13] for elliptic curve encryption. The computation overhead of the two schemes is shown in Fig. 3. The total number of concurrent signatures creation or verification is determined by the number of smart meters in the AMI network. It is obvious that our scheme achieves lower computation cost compared to the signcryption scheme.

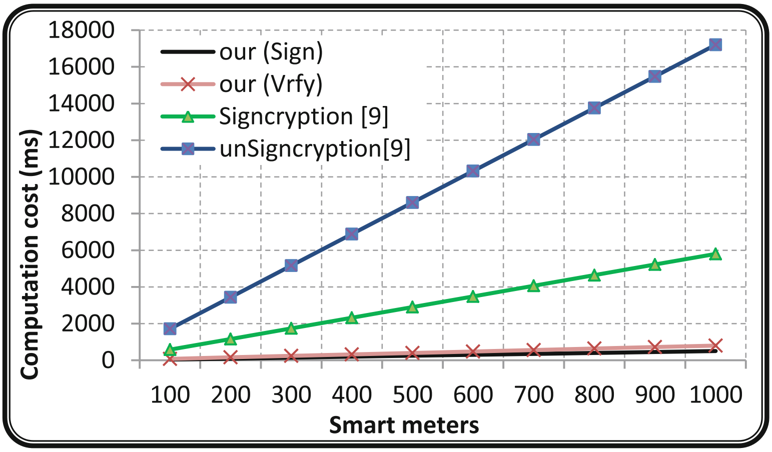


Fig. 3. Computation cost of the proposed scheme and the model in [9]

4.4 Ciphertext Size

Ciphertext (Ĉ) size is the size of: the encrypted data, signature and any additional cryptographic parameters attached to enable recovering the plaintext. In the proposed scheme, the ciphertext $\hat{C} = \left(\overline{M}, \sigma\right)$, where $\overline{M}$ is the encrypted data and σ is the ECDSA signature.

AES encryption does not enlarge data size; therefore M and $\overline{\mathrm{M}}$ both have the same size. The signature σ has two components (as per ECDSA details); hence the signature size is twice the length of the elliptic curve. Therefore, the ciphertext size is $|M| + |point|$. On the other hand, the ciphertext produced by the signcryption scheme presented in [9] is $\hat{C} = \left(C, C_{enc}, C_{sign}\right)$. C and C_{sign} are each twice the size of the

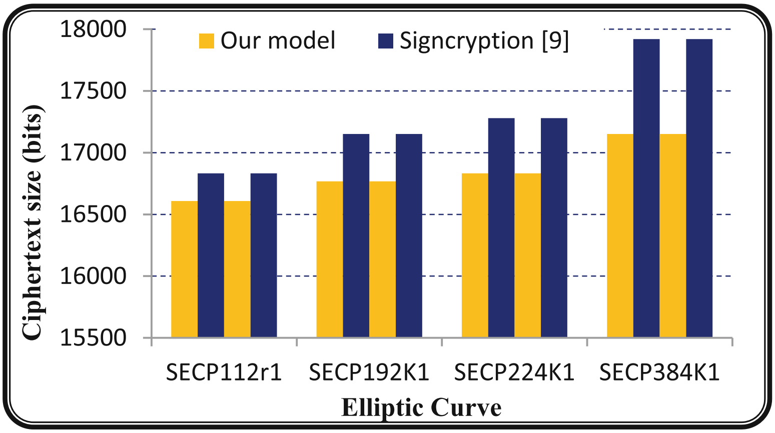


Fig. 4. Ciphertext size for different elliptic curves

elliptic curve as they are points on the curve. The scheme does not expand encrypted data too, therefore the size of the plaintext and C_{enc} are the same. The total length of the ciphertext under signcryption scheme will be $|M| + 2|point|$. Assuming a plaintext size of 16 KB, Fig. 4 illustrates the length of the resulting ciphertext under the proposed scheme and the one in [9] when using different elliptic curve standards.

5 Security Analysis

The proposed scheme provides encryption and authentication for downlink and uplink communication in AMI network. This section demonstrates the security analysis of the proposed scheme under the adversarial model presented in Sect. 3.2 by examining its resiliency against known attacks.

5.1 Passive Attacks

In order to guarantee data confidentiality, customer and supervisory node packets are never sent in clear, symmetric encryption algorithm *SYMM.ENC* (such as AES) is used to cipher those packets. Supervisory node (SCC) commands are encrypted using $k_{UD}, k_{MD}\, or\, k_{BD}$ depending on the transmission mode, while customer metering data is encrypted using k_{uu}. Adversary $\mathcal{A}$ who intercepts AMI communication channels will not be able to collect any useful information concerning customer behavior or usage pattern. He will not be able to identify the remote load commands issued by SCC, as well.

5.2 Impersonation Attack

Adversary $\mathcal{A}$ can impersonate any D_{LL} or D_{UL} if he manage to forge their signatures. Under the proposed chain based signature scheme, the public keys $\{PK_{UD}, PK_{MD}, PK_{BD}, PK_{UU}\}$ are used once and they are sent encrypted a priori. Thereby, it will be impossible for the adversary $\mathcal{A}$ to gather and cryptanalyze combinations of legitimate public keys/signatures for the purpose of forging valid signatures. Hence, the proposed scheme withstands impersonation attack.

5.3 Replay Attack

The adversary $\mathcal{N}_t \leq \mathcal{N}_{t-1}$ can capture and store valid network messages for the purpose of maliciously replaying them later. Such attack is easily detected in our scheme by using nonce. At any time, the received nonce should be greater than its predecessor; therefore replay attack is detected when $\mathcal{N}_t \leq \mathcal{N}_{t-1}$. It should be noted that the attacker can't predict the current nonce value as the initial nonce is generated randomly and is sent encrypted.

5.4 Message Modification Attack

ECDSA is a secure public key algorithm because it is computationally infeasible to modify the message $\overline{M}_t$ and its signature σ_t to construct a new message with valid signature. Therefore, $\boldsymbol{Vrfy}_{PK}\left(h_2\left(\overline{M}_t\right),\sigma_t\right)$ function will output zero if the message or the signature (or both) are altered in transit. Hence, the attached digital signature can serve as a guard against message alteration and the proposed scheme withstand against message modification attack.

6 Conclusion

In this paper, we have proposed a chain based signature scheme to provide authentic two-way communication in AMI network. The proposed scheme employs symmetric cryptography as well, in order to maintain data confidentiality. For optimal implementation of the proposed scheme, we have classified the AMI traffic into downlink and uplink, and we examined the transmission mode(s) required by each class. We have shown that the proposed scheme can resist various known attacks and is efficient in terms of the computation overhead and the ciphertext length.

References

1. Abdulrahman, Y., Saifur, R.: Smart grid networks: promises and challenges. JCM **7**(6), 409–417 (2012)
2. Gungor, V.C., et al.: A survey on smart grid potential applications and communication requirements. IEEE Trans. Ind. Inform. **9**(1), 28–43 (2013)
3. NIST framework and roadmap for smart grid interoperability standards release 1.0 (2010)
4. Massoud, A.: A smart self-healing grid: in pursuit of a more reliable and resilient system [in my view]. IEEE Power Energy Mag. **12**(1), 110–112 (2014)
5. Hayden, K.-H. S., Sammy, H.M.K., Edmund, Y.L., King-Shan, L.: Zero-configuration identity-based signcryption scheme for smart grid. In: IEEE International Conference on Smart Grid Communications, October 2010
6. Debiao, H., Huaqun, W., Muhammad, K.K., Lina, W.: Lightweight anonymous key distribution scheme for smart grid using elliptic curve cryptography. IET Commun. **10**(14), 1795–1802 (2016)
7. Jia-Lun, T., Lo, N.-W.: Secure anonymous key distribution scheme for smart grid. IEEE Trans. SmartGrid **7**, 906–914 (2015)
8. Saxena, N., Grijalva, S.: Efficient signature scheme for delivering authentic control commands and alert messages in the smart grid. IEEE Trans. Smart Grid **9**, 4323–4334 (2017)
9. Alharbi, K., Lin, X.: Efficient and privacy-preserving smart grid downlink communication using identity based signcryption. In: 2016 IEEE Global Communications Conference, Washington, DC, pp. 1–6 (2016)
10. Libert, B., Quisquater, J.J.: New identity based signcryption schemes from pairings. In: IEEE Information Theory Workshop, Paris, France (2003)
11. Lal, S., Kushwah, P.: ID based generalized signcryption, Cryptology ePrint Archive http://eprint.iacr.org/2008/84 (2008)

12. Mahmood, K., Chaudhry, S.A., Naqvi, H., Kumari, S., Li, X., Sangaiah, A.K.: An elliptic curve cryptography based lightweight authentication scheme for smart grid communication. Future Gener. Comput. Syst. **81**, 557–565 (2018). https://doi.org/10.1016/j.future.2017.05.002
13. SEC 2: Recommended Elliptic Curve Domain Parameters, Certicom Research, 20 September 2000