Efficient authentication scheme for data aggregation in smart grid with fault tolerance and fault diagnosis

Citation	Li, Depeng et al. "Efficient Authentication Scheme for Data Aggregation in Smart Grid with Fault Tolerance and Fault Diagnosis." Proceedings of the Innovative Smart Grid Technologies (ISGT), 2012 IEEE PES (2012). 1–8.
As Published	http://dx.doi.org/10.1109/ISGT.2012.6175680
Publisher	Institute of Electrical and Electronics Engineers (IEEE)
Version	Author's final manuscript
Accessed	Sun Mar 12 03:17:46 EDT 2017
Citable Link	http://hdl.handle.net/1721.1/77999
Terms of Use	Creative Commons Attribution-Noncommercial-Share Alike 3.0
Detailed Terms	http://creativecommons.org/licenses/by-nc-sa/3.0/

Efficient Authentication Scheme for Data Aggregation in Smart Grid with Fault Tolerance and Fault Diagnosis

Depeng Li, Zeyar Aung, John R. Williams, and Abel Sanchez

***Abstract*— Authentication schemes relying on per-packet signature and per-signature verification introduce heavy cost for computation and communication. Due to its constraint resources, smart grid's authentication requirement cannot be satisfied by this scheme. Most importantly, it is a must to underscore smart grid's demand for high availability. In this paper, we present an efficient and robust approach to authenticate data aggregation in smart grid via deploying signature aggregation, batch verification and signature amortization schemes to less communication overhead, reduce numbers of signing and verification operations, and provide fault tolerance. Corresponding fault diagnosis algorithms are contributed to pinpoint forged or error signatures. Both experimental result and performance evaluation demonstrate our computational and communication gains.**

***Index Terms*—Authentication, batch verification, digital signature, fault tolerance, fault diagnosis, smart grids.**

I. Introduction

THIS decade is going to witness the paradigm shift from the traditional electrical power grid to the smart grid around the globe. The smart grid is a network of smart devices (e.g. commodity computers) and power infrastructure [7]. Smart grid can satisfy power demands in real-time, optimally transmit and distribute electricity from suppliers to consumers, and automatically monitor the power usage status via integrating current communication and information technologies [1]. Significant benefits, including improved energy efficiency, promoted power reliability and decreased carbon emission, can be provided by smart grid [6].

A. Security for Data Aggregation in Smart Grid

To achieve goals aforementioned, new components such as smart meters, two-way communication networks, monitoring system, decision-making intelligent system, etc. are involved in smart grid, which inevitably introduces new security risks related to data collection and processing, data transportation, automation control and system monitoring. Communication system, for example, utilizing Zigbee [2], or Wi-Fi alliance is vulnerable to eavesdropping, unauthorized participation and modification, Denial-of-Service (DoS) attacks and other malicious activities while sending data on the air.

This paper mainly focuses on security of power usage data aggregation. Smart meters manage and control electricity flows to and from the end customers, also record customers' power usage data in real-time and periodically report them to the collector devices in the neighborhood as well [7]. Collectors, in turn, send the collected data to control centers of utility companies to support the pricing and decision-making. Not all smart meters can communicate with the collector directly. Intermediate smart meters cooperate in relaying packets on behalf of one another till packets reach the collector. This procedure is called *data aggregation*. The security concern arises: the intermediate node can drop, modify, eavesdrop, and forge data during data aggregation without been recognized by the smart grid. Therefore, security schemes are mandatory.

The proposed security scheme's performance is seriously dependent on the number of messages that the scheme processes and protects. It is estimated that the amount of data transported across the smart grid will be an increase of an order of magnitude [7]. This introduces substantial cost increment for security mechanism. Increased data belongs to different kinds of categories [21]. Therefore, different security requirements are required. Protective relaying data, for instance, required to be transported in real-time without loss and any latency for them with more than 4 milliseconds is not affordable. In contrast, meter reading, for example, can be tolerant to latency ranging from minutes to hours [5]. Consequently, various security solutions should be adapted to meet varying requirements. Moreover, because of the huge amount of data, limited communication capacity, and low-capacity devices of smart meters, security solutions are supposed to be lightweight in terms of computational cost and communication overhead.

B. Our Contributions

The nature of smart grid ranks safety, availability and reliability the highest priority for smart grid. In contrast, Confidentiality, Integrity and Authentication (CIA), security requirements that are ranked as the highest priority in Information Technology (IT) networks, are degraded at the second position for smart grid. Therefore, security approaches should not only satisfy CIA, but, more importantly, heavily emphasize the built-in availability as well as minimize its fault

Depeng Li and Zeyar Aung are with Masdar Institute of Science and Technology, Abu Dhabi, United Arab Emirates (e-mails: {dli, zaung}@masdar.ac.ae).

John R. Williams and Abel Sanchez are with Auto-ID Lab, Massachusetts Institute of Technology, Cambridge, MA 02139, USA (e-mails: {jrw, doval}@mit.edu).

/ offline time during smart grid system executions by providing fault-tolerance and fault analysis.

To secure data aggregation, pioneer works have been extensively explored [10]-[21] and they mainly focus on the integrity and confidentiality; nevertheless, authentication and its efficiency are disregarded. Without authentication service, malicious smart meters, insiders or outsiders, could pretend to be someone else' meters, spoof the forgery customer ID, falsify power usage data, and get free power. Customers face the risk of financial loss. Furthermore, due to extensive requests for availability, the authentication approach needs to be fault-tolerant in architecture so that it still functions even when the collector is offline. Moreover, fault diagnosis should be highlighted. Otherwise, the fault time cannot be minimized. At last, schemes such as pairwise keys or group keys show complicated configurations or lack non-repudiation feature, respectively. They both demonstrate hidden costs to bootstrap themselves. Authentication schemes relying on public key operations are computational heavy. Since most smart meters lack powerful process capability, diminishing the number of digital signatures and verifications is highly demanded.

This paper will propose an efficient authentication scheme for power usage data aggregation in Neighborhood Area Network (NAN). Contributions are listed below:

1) **Fault tolerance architecture**: this approach deploys digital signature so that when the collector is out of service, alternative or backup collector can execute the authentication approach without any additional configuration or setup. This fixes the single-point failure. Furthermore, it transports aggregation data via Minimum Spanning Tree (MST), and rearranges MST whilst some smart meter nodes cannot response;
2) **Efficiency**: to reduce the number of signature and verification operations, signature amortization and batch verification are presented, respectively. To decrease the number of signatures sent on channels, MST-based signature aggregation tree is proposed;
3) **Fault diagnosis**: diagnose tools are proposed for batch verification and signature aggregation to detect failure points and to minimize the whole fault time. Erasure code is combined with signature amortization to support fault tolerance against signature packet loss in harsh communication environment of smart grid;

The rest of this paper is organized as follows. Background information is introduced in Section II. Related works are reviewed in Section III. Our proposed solution is described at Section IV. Security analysis, performance evaluation and experimental results are presented in Section V. Concluding remarks are given in Section VI.

II. Background and Basic Schemes

In this section, we briefly review the smart grid, as well as the specific security requirements concerning with smart grid.

A. Smart Grid and its Security Requirements

As shown in Fig. 1, smart grid is comprised of *power generation*, *power transmission* and *power distribution* components which produce power in bulk quantities, carry power over long distances, and distribute electricity to end consumers in local, respectively [7]. New technologies and components are included: Smart meters utilize Phasor Measurement Units (PMU) and Global Positioning System (GPS) time stamps to measure power status and electricity consumption based on waveforms as well as the magnitude & phase angle of voltage [12]. Status and usage data is collected and information flow is transferred following the path from smart meters, collectors, sub-stations, and control centers of utility companies. Meanwhile, demand-respond messages are sent back and forth in the same way but bi-directionally.

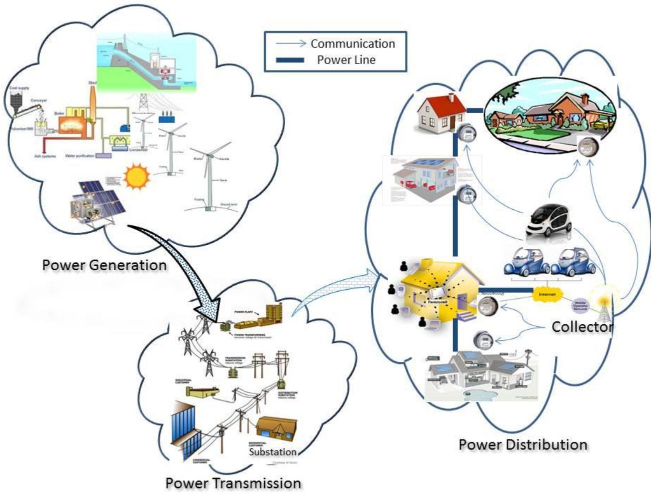


Fig. 1. Architecture of Smart Grid

Two-way communication system, wired or wireless, acts as the backbone to relay packets in smart grid. To connect substations and control centers, wired networks such as Power Line Communication (PLC) [5]. are deployed to transfer control and usage data. In case of communications for the last mile, wireless networks such as Zigbee, Wi-Fi alliance, etc. are preferred in favor of the cost savings in NAN. Zigbee / 802.15.4 polls data every 15 minutes and standardizes with the data rate of 250kbps in maximum, which both result in a 4 milliseconds' interval. This shows constraint communication bandwidth [2]. Wi-Fi alliance demonstrates higher communication capacity; nevertheless, it does not have the mesh function, yet.

Communication systems of smart grid face the harsh network environment, unreliable wireless channels and relatively strict security requirements. Resources-constraint characteristics make it more difficult to secure smart grid system: 1) although utilizing wireless network communication in smart grid saves the cost, the native infrastructure is unsecure; 2) most smart meters are configured with low-capacity devices which tends to be restricted in their computational capability and cannot perform many and frequent computational-intensive operations such as public key cryptographic operations. Furthermore, limited communication bandwidths and uncertainly channel leads to packet loss / error as well; 3) the smart metering system's open architecture shares the wireless medium and channels. It is easier for illegitimate users and malicious adversaries to access, interfere, or block wireless channels. As a result, damages, such as secret information leak or high rate of packet loss / error, are more likely to happen.

Electric power industries require that power delivery should be always available without any human hurts [1]. Therefore, in smart grid, the availability, safety, and reliability are the most important security objectives with higher precedence over CIA. In this paper, following the priority preferences, the proposed authentication scheme will not only authenticate data aggregation but robust its availability by providing fault-tolerance and fault diagnosis services.

TABLE I
SAMPLES OF TIMES LATENCY AND TYPE STYLES

Max. Latency	**Comm. Type**
$\leq$ 4 ms	Protective relaying
Sub-seconds	Status Monitoring
Seconds	Substation SCADA
Minutes	Market Pricing Info
Hours	Meter Reading
More than Days	Long-term usage

As mentioned earlier, there are different kinds of data sent in smart grid which has different requirements. Table I [5] shows data categories and corresponding maximum latencies across smart grid system. The acceptable latencies for protective relaying, status monitoring and substation SCADA are less than 4ms, sub-seconds and seconds, respectively. Meanwhile, they are required to be delivered without any loss. So, it is possible that data (e.g. meter reading) protected by our approach is dropped while the bandwidth is inadequate. Our proposal should handle scenarios that authentication message is lost and will not be re-transported again.

B. Digital Signature Schemes

Digital signature, a cryptographical primitive is utilized in this paper to authenticate data aggregation. As one example, we introduce a short signature scheme of Boneh, Lynn, and Shacham (BLS) [27], a bilinear map in this section. Then, we briefly describe batch verification [24][27][28] and signature aggregation [29]. Notice that digital signatures in integer cryptography and Elliptic Curve cryptography [9] can be used to replace bilinear map in our proposal without any modification. Next, we use Public Key Infrastructure (PKI) [9] [31] and Trusted Third Party (TTP) to issue / revoke certificates and public & private key pairs for smart meters.

1) Bilinear map

Bilinear map [30] works as the basis of our approach. $\mathbb{G}$ and $\mathbb{G}_{\mathbb{T}}$ are a cyclic additive group and a cyclic multiplication group generated by P with the same order q, respectively. A mapping $\hat{e}: \mathbb{G} \times \mathbb{G} \longrightarrow \mathbb{G}_{\mathbb{T}}$ satisfies the following properties:

- **Bilinear:** for all $u, v \in \mathbb{G}; a, b \in \mathbb{Z}$, we have $\hat{e}(u^a, v^b) = \hat{e}(u, v)^{ab}$, where $=$ is an equation;
- **Computable**: there exists an efficient computable algorithm to compute $\hat{e}(u, v), \forall u, v \in \mathbb{G}$;
- **Non-degenerate**: for the generator g of $\mathbb{G}$, p is the order of $\mathbb{G}$, we have $\hat{e}(g, g) \neq 1 \in \mathbb{G}_{\mathbb{T}}$;

2) BLS short signature scheme

Key generation -

- Randomly selects $x \overset{R}{\leftarrow} \mathbb{Z}_p$ and calculates $y \longleftarrow g^x \in \mathbb{G}$; x and y are the private and public keys respectively; $x \in \mathbb{Z}_p$ *and* $y \in \mathbb{G}$; ← is assignment;

Signature generation – Sender Alice calculates signature:

- Given a message $M \in \{0,1\}^*$, computes $h \longleftarrow H(y, M)$ where H is a collision-resistant hash function e.g. *MapToPoint* hash [27] [28] such that $H: \{0,1\}^* \rightarrow \mathbb{G}$;
- Computes $\sigma \leftarrow h^x$ where $\sigma \in \mathbb{G}$ is signature;

Signature verification – Receiver Bob verifies signature:

- Obtains Alice's public key y, signature σ and message M, performs $h \longleftarrow H(y, M)$;
- Performs verification: $\hat{e}(h, y) = \hat{e}(\sigma, g)$;

3) Batch Verification

Signature generation – Sender Alice calculates signatures:

- Given n messages $\{M_1, M_2, \cdots M_n\}$ and public key y, computes $h_i \longleftarrow H(y, M_i)$ where $M_i \in \{0,1\}^*$ for all $i \in [1, n]$;
- Computes $\sigma_i \leftarrow {h_i}^x$ where $\sigma_i \in \mathbb{G}$ is signature;

Batch verification – Receiver Bob verifies signature:

- Obtains Alice's public key y, signatures $\{\sigma_1, \sigma_2, \cdots \sigma_n\}$ and messages $\{M_1, M_2, \cdots M_n\}$; Calculates $h_i \longleftarrow H(y, M_i) \in \mathbb{G}$;
- Performs verification:

$$\hat{e}\left(\prod_{i=1}^{n} h_i, y\right) = \hat{e}\left(\prod_{i=1}^{n} \sigma_i, g\right) \qquad (1)$$

4) Signature aggregation

Signature aggregation

- Distinct n users $\{u_1, u_2, \cdots u_n\}$ sign n distinct messages $\{M_1, M_2, \cdots M_n\}$ with its own public key, $\{y_1, y_2, \cdots y_n\}$ by BLS scheme, respectively: Calculate: $h_j \longleftarrow H(y_{j,} M_j) \in \mathbb{G}$; $1 \leq j \leq n$ Obtains signatures: $\{\sigma_1, \cdots \sigma_n\}$ where $\sigma_j \longleftarrow {h_j}^{x_j} \in \mathbb{G}$ is calculated by corresponding meter;
- Aggregates all n signatures into a single signature $\sigma_{1,2,\cdots n} \longleftarrow \prod_{j=1}^{n} \sigma_j \in \mathbb{G}$;

Signature verification – Verifier can:

- Obtains n users' Public keys, $\{y_1, y_2, \cdots y_n\}$; One signature aggregations $\{\sigma_{1,2,\cdots,n}\}$; Messages $\{M_1, M_2, \cdots M_n\}$; Calculate $h_j \longleftarrow H(y_{j,} M_j) \in \mathbb{G}$
- Performs verification:

$$\hat{e}(\sigma_{1,2,\cdots,n}, g) = \prod_{j=1}^{n} \hat{e}(h_j, y_j) \qquad (2)$$

5) Other authentication solutions

In addition to the asymmetric key solution (e.g. digital signature), symmetric cryptographic primitives [9], such as pairwise key and group key, can also be deployed to authenticate aggregation data. However, group key lacks the non-repudiation feature, a critical component for fault diagnosis. Pairwise key faces the single-point failure: in general, pairwise keys are configured between the collector and every smart meter. Once the collector is out-of service, pairwise key solution fails. Since availability plays a key role in smart grid and fault diagnosis is an elementary part for our proposal, either group key or pairwise key cannot be utilized.

III. Related Works

Since smart grid is a new area in both industry and academic, security research for smart grid is just starting [21]. A few pioneer works [11]-[20] provide implementations / schemes but they mainly focus on integrity and confidentiality rather than authentication. So far, to provide authentication via public cryptography, recent approaches proposed for smart grid still utilize per-packet signature and per-signature verification. Meanwhile, a number of researches [8] have been conducted for lightweight authentications but they cannot be directly used in smart grid due to smart grid's unique requirements. In this section, we will review security approaches designed to protect smart grids.

A. Bartoli *et al.* [17] propose a secure and lossless aggregation protocol providing CIA service. This protocol assumes that smart meters connect to a gateway via a tree topology in which data can be aggregated from leaves to the root. As a sophisticated solution, it includes two security solutions, end-to-end and hop-by-hop. In the former, based on a shared secret between the gateway and every meter, the gateway can find / derivate the key to decrypt the ciphertext, check its integrity, and verify its authentication for every data/packet sent from the meter to the gateway. In the later, pairwise keys are used between each smart meter and its one-hop neighbor to achieve the CIA. The pairwise key plays the authentication role. Nevertheless, the cost to establish presumed pairwise keys between every meter and the gateway or every two meters is hidden. Furthermore, key maintenance cost also needs to be included: when the legal / broken meters are replaced with upgraded / backup ones, new meters are added, malicious meters are expelled or mobile meters are roaming, extra cost should be spent to establish pairwise keys. Rules and costs to refresh pairwise keys are also needed.

F. Li, B. Luo and P. Liu [16] present an efficient information aggregation approach, in which, a aggregation tree constructed via breadth-first traversal of the graph and rooted at the collector unit, is deployed to cover all smart meters in the neighborhood. Aggregation information should be transported, in the tree, from child nodes to their parent node one hop by another and finally reach the root. This protocol can let the control unit collect all smart meters' information in this area. Furthermore, to protect users' privacy, all information is encrypted by homomorphic encryption algorithm. Since no authentication scheme is emphasized, the approach faces the potential risk that malicious smart meters can forge packets but smart grid system cannot detect / diagnose bogus data.

D. Wu and C. Zhou [15] propose a key management scheme. In order to achieve authentication goal, well-known Needham-Schroeder protocol is deployed to get the session key. In sake of computation and communication gain, general / integer public key cryptography is replaced with the elliptic curve one. This proposal eliminates main-in-the-middle and replay attacks effectively. The one-time use rule and on-the-fly key generation scheme are also emphasized in the proposal to shield the vulnerability on both communication keys and sessions keys.

H. Khurana *et al.* [22] provide guidelines for authentication protocol used in smart grid. Seven principles, including names, encodings, trust assumptions, secret releases, security parameters, etc. are proposed.

D. Wei *et al.* [11] propose a distributed and scalable security framework in concept with the layered architecture. It can protect the smart grid against attacks from either Internet or internal network via integrating security agents, security switches and security managements.

J. Zhang and C. A. Gunter [18] propose an approach to secure multicast in smart grid via deploying IPsec protocol and Group Internet Key Exchange (GIKE). Both IPsec and GIKE are standards for IT network; nevertheless, they are not specifically designed for multicast in smart grid.

A. R. Metke and R. L. Ekl [13] propose a security solution for smart grid utilizing the PKI and trust computing. The three components, certificate management, trust anchor security and attribute certificate in PKI are carefully illustrated and tailored to meet smart grid's security requirement.

J. Chao *et al.* [20] adapts RFID communication standard security protocol and utilizes it in smart grid. One-time password is deployed for user authentication.

H. K.-H. So *et al.* [19] propose an Identification-Based Signcryption (IBS) approach based on elliptic curve public key cryptography to provide CIA services.

IV. Proposed Solutions

In this section, we propose an efficient and robust authentication approach to legalize the data aggregation with tremendously less signing and verification operations. Meanwhile, the utilization of signatures and the providing of fault-tolerance & fault-diagnosis services satisfy the requirement for high availability in smart grid.

Confidentiality also plays a critical role in smart grid communication security. It can be achieved by using encryption algorithms (e.g. AES [36], FEA-M [33], ID-based pairing [28], etc.) and key agreement (e.g. Diffie-Hellman key agreement [9], group key agreement [34], etc.). The scope of this paper is limited to authentication solutions.

A. MST

Fig. 2(a) is an example of the smart grid's NAN including both the collector and a number of smart meters. It can be denoted as $M = (\{m_1; \cdots; m_{|M|}\}; L)$ where m_i is a smart meter node and L is the set of communication channels established by two smart meters. So, M can be modeled as a connected, undirected graph $G = (\{n_1; \cdots; n_{|M|}\}, E, W(e))$ where vertex n_i corresponds to node m_i in M, E denotes the set of edges in L and W is the set of weights for all edges. There is an edge in E between a pair of vertices n_i and n_j if nodes m_i and m_j in M enable successful communication directly. Each edge is associated with a weight number, calculated by the combination of communication bandwidth as well as both smart meters' CPU power, memory capacity, etc. Our solution utilizes MST algorithm (please refer to [32] for details) to construct a spanning tree, based on which, data is aggregated. Fig. 2(b) demonstrates the result.

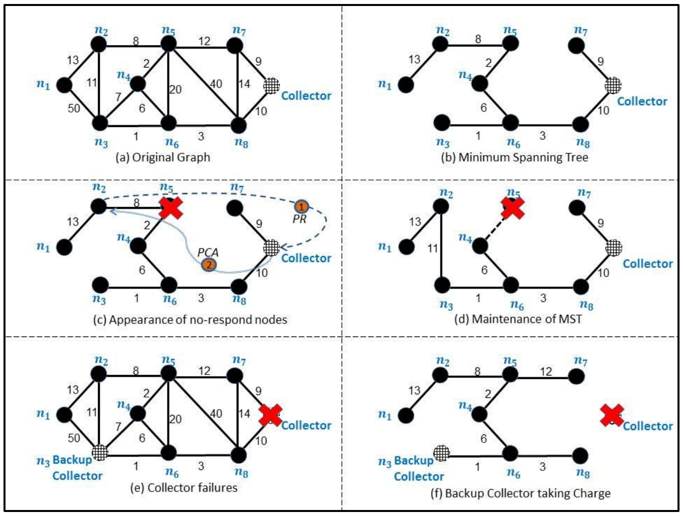


Fig. 2. Example of NAN constructed with smart meters and collectors

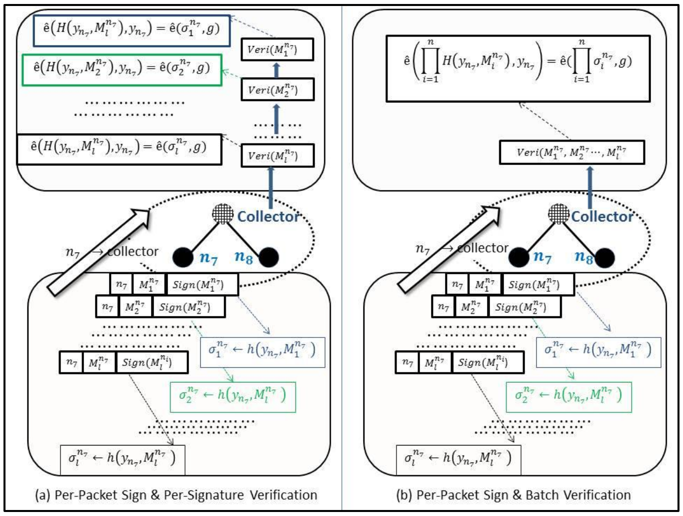


Fig. 3. Batch Verification

We notice that there are some non-respond scenarios for some smart meters: 1) to save the energy, smart meters switch to sleep-mode in case of no power usage. 2) smart meters are out of service because of hardware / software failures. 3) smart meters do not respond to any request due to overloaded tasks or malicious activities such as physical tampering, DoS attacks, etc. Approach in [16] using Breadth-First Searching (BFS) spanning tree does not mention how to maintain the spanning tree in case of scenarios aforementioned. In our solution, when not receiving keep-alive/beacon messages from their parents, smart meters send out *Parent Request* (PR) to collector. The collector re-executes MST algorithm within itself, and after completion, broadcasts *Parent-Child Association* (PCA) updates to nodes with either parent or children node changes. When the collector fails, backup collector takes charges seamlessly via constructing MST and broadcasting PCAs to all smart meters in NAN. It collects aggregated data via MST and verifies corresponding signatures without extra configurations. The single-point failure problem is fixed. Less communication cost is required to maintain MST. Please refer to Fig. 2(c) - Fig. 2(f) for details.

B. Batch Verification and Trinary Diagnose Tree

The deployment of digital signature makes our solution fault tolerant in terms of architecture. However, per-packet signing and per-signature verification is computationally expensive. The collector needs to verify all meters' signatures. Batch verification provides the same level of security but reduce the number of verification operations from l to 1 when verifying l signatures signed by the same sender.

Fig. 3 (a) and (b) demonstrate our discussion for batch verification. Since a pairing operation costs significantly higher than multiplication [30], batch verification saves immense CPU processing resources for collectors.

Fault diagnosis Algorithm

To pinpoint the cause of batch verification failures and locate bogus signatures, we contribute $\alpha - ary$ tree-based fault diagnosis verification algorithm. Since, in case that $\alpha = 3$, the minimum number of verifications is invoked, trinary verification tree is used. Comparing with [23], our proposal is more efficient.

Algorithm 1: *Batch Verification*

```
/* Collector has smart meter, $n_i$'s public key $y_{n_i}$ and $l_i$, (the number of accumulated signatures from $n_i$) in advance. */
Collector processes the followings:
$temp_h \leftarrow 1$;  $temp_\sigma \leftarrow 1$;
For  ( k←1;   $k \le l_i$;   k← $k+1$ )
    Listens on the channel and receives triple { $n_i, M_k^{n_i}, \sigma_k^{n_i}$ }
    $temp_h \leftarrow temp_h \times H(y_{n_i}, M_k^{n_i})$;
    $temp_\sigma \leftarrow temp_\sigma \times \sigma_k^{n_i}$;
Endfor
/* Verifies $l_i$ signatures sent from smart meter $n_i$ */
If  $\hat{e}\,(temp_h,\ y_{n_i}) \neq \hat{e}\,(temp_\sigma,\ g)$
    calls  trinity tree-based fault diagnosis verification alg.;
    return FALSE;
Endif
Return TRUE;
```

The collector constructs a trinary verification tree T in which every node can be denoted as $< h,\ i >$ where h is the height (level) of the node and i is the index of the node at level h. Thus, every node is identified uniquely. Each node is associated with a signature. There are two kinds of nodes in T, leaf nodes and intermediate nodes. The leaf node's signature is assigned with signatures the collector receives from the sender. The intermediate node has three children and the signature associated with it is the multiplication of all its children's signatures. Please refer to (3) on how to calculate signatures.

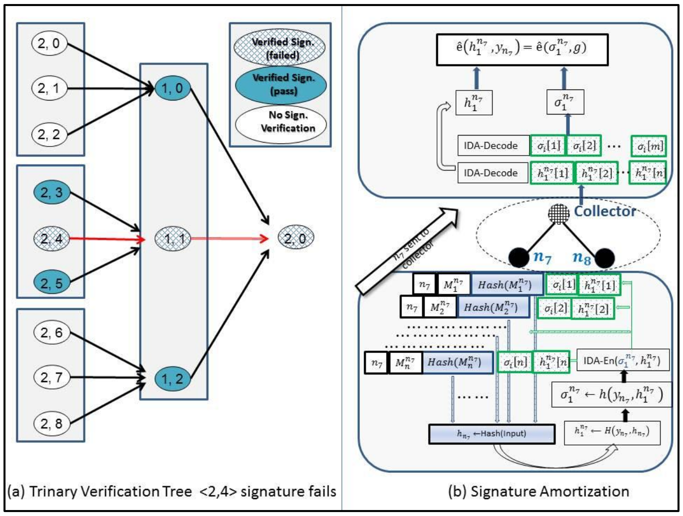
$$\sigma_{<h,i>} \leftarrow \begin{cases} \sigma_i & if\ n_{<h,i>} = Leaf \\ \prod_{l=0}^{l\le 2} \sigma_{<h+1,3i+l>} & if\ n_{<h,i>} \neq Leaf \end{cases} \quad (3)$$

The fault diagnosis verification algorithm follows the breadth-first travel algorithm: starting at root, if there is a verification failure node, all its children will be verified. Otherwise, there is no need to verify any of its offspring. Repeat this procedure till the breadth-first travel algorithm completes. Please refer to Fig. 4 (a) as an example.

C. *Signature Amortization for Package Blocks*

Batch verification scheme can verify *n* signatures in one verification operation rather than per-signature individually. This saves significant processing resources for collectors. However, the smart meter still has to sign per-packet individually. Since most smart meters are low-capacity, too many signings will drain smart meter's processing capacity. Solutions to sign *n* packets with only one signature are highly demanded by smart meters. Our approach in this subsection deploys an efficient scheme, Signature Amortization (SAm) to amortize the digital signature over a block of packages.

Furthermore, the communication in NAN of smart grid has channel instability and restricted resources. Packets could be lost during data communications. Moreover, power control messages such as demand-respond data own higher priority over power usage data. When the congestion happens, it is possible to drop the power usage data and its signature. Therefore, if a signature lost, a block of packages cannot be verified. Erasure code such as Information Dispersal Algorithm (IDA) [25] could be used to encode a signature and amortize the result over a block of packets. Even only *m* out of *n* packets ($m < n$) are successfully delivered to the receiver end, the signature still can be decoded. SAm is described in Alg. 2. For detailed implementation of IDA, please refer to [34] and [25]. Fig. 4(b) illustrates this process.


2, 0
2, 1
2, 2
2, 3
2, 4
2, 5
2, 6
2, 7
2, 8
1, 0
1, 1
1, 2
2, 0
Verified Sign. (failed)
Verified Sign. (pass)
No Sign. Verification
$\hat{e}(h_1^{n_7}, y_{n_7}) = \hat{e}(\sigma_1^{n_7}, g)$
$h_1^{n_7}$
$\sigma_1^{n_7}$
IDA-Decode
$\sigma_i[1]$ $\sigma_i[2]$ ⋯ $\sigma_i[m]$
IDA-Decode
$h_1^{n_7}[1]$ $h_1^{n_7}[2]$ ⋯ $h_1^{n_7}[n]$
Collector
n_7 sent to collector
n_7
n_8
n_7 $M_1^{n_7}$ $Hash(M_1^{n_7})$ $\sigma_i[1]$ $h_1^{n_7}[1]$
n_7 $M_2^{n_7}$ $Hash(M_2^{n_7})$ $\sigma_i[2]$ $h_1^{n_7}[2]$
n_7 $M_n^{n_7}$ $Hash(M_n^{n_7})$ $\sigma_i[n]$ $h_1^{n_7}[n]$
IDA-En($\sigma_1^{n_7}, h_1^{n_7}$)
$\sigma_1^{n_7} \leftarrow h(y_{n_7}, h_1^{n_7})$
h_{n_7} ←Hash(Input)
$h_1^{n_7} \leftarrow H(y_{n_7}, h_{n_7})$

(a) Trinary Verification Tree <2,4> signature fails

(b) Signature Amortization

Fig. 4. Signature Amortization

```
Algorithm 2: Signature Amortization (SAm)
Each smart meter $n_i$ signs a block of messages $M_1, M_2 \cdots M_{n_i}$
$h_{n_i} \leftarrow NULL$ which is used to store hash result of the block
For ( k←1;   $k \le n_i$;   k← $k + 1$ )
        $h_{n_i} \leftarrow h_{n_i} \;||\; Hash(M_k)$
Endfor
$h_{n_i} \leftarrow Hash(h_{n_i})$;   $h_i \longleftarrow H(y_{n_i}, h_{n_i})$ ;   $\sigma_i \leftarrow h^{x_i}(h_i)$;
$\sigma_i[1], \sigma_i[2], \cdots \sigma_i[n_i] = IDA - en(\sigma_i, n_i, m_i)$ /* Get n slices */
$h_1[1], h_1[2] \cdots h_1[m] = IDA - en(h_i, n_i, m_i)$ /* Get n slices */
For ( k←1;   $k \le n_i$;   k← $k + 1$ )
    Sends to $n_i$'s parent node: { $M_k \;||\; \sigma_i[k] \;||\; h_1[i]$ }
Endfor
Collector receives m slices { $\sigma_i[1], \sigma_i[2], \cdots \sigma_i[m_i]$ } and
        m block hash result slices: { $h_1[1], h_1[2] \cdots h_1[m]$}
Reconstructs signature:  $\sigma_i$ ←IDA-de($\sigma_i[1], \sigma_i[2], \cdots \sigma_i[m_i]$)
Recalculates block hash: $h_1$ ← IDA-de($h_1[1], h_1[2] \cdots h_1[m]$)
Performs verification:   $\hat{e}(h_1, y_{n_i}) = \hat{e}(\sigma_i, g)$
```

D. *MST-based Signature Aggregation (MST-SA)*

In batch verification and signature amortization, all signatures will be sent over aggregation path and they arrive at the collector. Considering about the length of a signature (e.g., 1024 bits for RSA and 157 bits for pairing [27]), it will consume the limited bandwidth of wireless communication. Signature aggregation can save communication cost via aggregating a number of signatures into a single one, only which will be transported on the air.

MST-SA: Our proposal, MST-SA integrates MST structure with the signature aggregation scheme. We term it *MST signature tree* (shortly, signature tree) which holds the same nodes and structure as MST. Furthermore, each node, namely n_i, representing a smart meter, is associated with two signatures, $(\sigma_{n_i}^k, \sigma_{n_i,tree}^k)$. The former, $\sigma_{n_i}^k$ (namely node signature) is the signature to sign message m_k from the smart meter n_i (The collector's node signature is assumed to be *1*). The later, $\sigma_{n_i,tree}^k$, (namely tree signature) is the signature for sub-tree rooted at node n_i. $\sigma_{n_i,tree}^k$ is calculated with signature aggregation scheme – multiplying n_i's node signature $\sigma_{n_i}^k$ by all tree signatures of node n_i's children nodes. Please refer to (4) as follows:

$$\sigma_{n_i,tree}^k \leftarrow \begin{cases} \sigma_{n_i}^k, & if\ n_i\ =\ leaf\ node \\ \sigma_{n_i}^k \prod_{x=0}^{x \le l} \sigma_{n_x,tree}^k, & if\ n_i\ \ne\ leaf\ node \end{cases} \tag{4}$$

where $\forall n_x\ is\ n_i^{\prime}s\ child;\ k:\ message\ number$

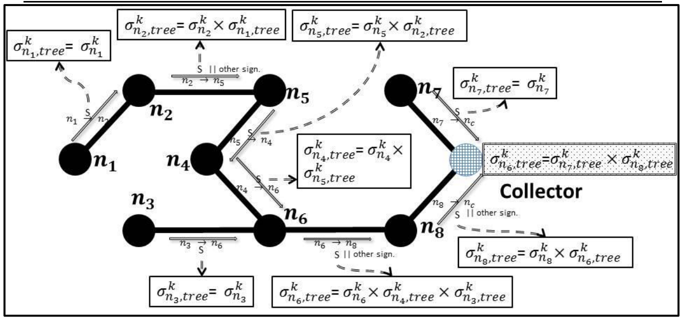
In the MST-based data aggregation, after receiving all children nodes' tree signatures, a node, following (4), calculates its tree signature which is sent to its parent node. Then, its parent follows the same process. Repeat this procedure with the bottom-up manner. At last, tree signature of the root node can be calculated by the collector. Alg. 3 and Fig. 5 demonstrate MST-based Signature Aggregation.

An active attacker can drain the collector's computational resource by constantly sending bogus signatures. To handle the scenario that the verification for root node's tree signature fails, MST-based fault diagnosis verification algorithm is designed to pinpoint the forged signings for signatures aggregation operation. The collector asks all nodes in MST for their tree signatures. After receiving them all, the collector constructs the MST-based signature tree, follows the post-order tree travel algorithm to explore every node in it, and verify every node's tree signature. If there is a failed signature from any smart meters, this algorithm will not calculate any ancestor of this failed node until meeting a leaf node during post-order tree travel. All this procedure is described in Alg. 4.

```
Algorithm 3: MST-based Signature Aggregation
Every node  $n_i$ in MST (except collector):
Calculates its node signature  $\sigma_{n_i}^k \leftarrow H^{x_{n_i}}(y_{n_i}, m_k)$ via BLS
IF $n_i \ne$ leaf node
    $\sigma_{n_i,tree}^k \leftarrow 1$
```

```
        Loop until receive all child nodes' signatures
            Listens on the channel;
            Receives { $n_{child-x}$ || $m_k$ || $\sigma^k_{n_{child-x}}$ }
            Calculates $\sigma^k_{n_i,tree} \leftarrow \sigma^k_{n_i,tree} \times \sigma^k_{n_{child-x}}$
        End Loop
    Else    tree signature $\sigma^k_{n_i,tree} \leftarrow \sigma^k_{n_i}$
    EndIF
    Sends { $n_i$ || $m_k$ || $\sigma^k_{n_i,tree}$ } to $n_i$'s parent node in MST;
```

```
Algorithm 4: MST-based fault diagnosis verification Alg.
for Collector:
Asks all nodes in MST for sub-tree signatures and wait to receive them all
Constructs the MST signature tree
Loop post-order tree travel algorithm to assign received sub-tree signatures to corresponding nodes
    IF    ($n_i$ = leaf nodes)
          $\sigma^k_{n_i,tree} \leftarrow \sigma^k_{n_i}$
    Else  $\sigma^k_{n_i,tree} \leftarrow corresponding\ subtree\ signature$
    EndIF
End Loop
Loop (every node in MST is visited by Post-order tree travels)
    IF    ($n_j$ = leaf node)
          verification $ê(h_i, y_i) = ê(\sigma^k_i, g)$
    Else  verification:
          $ê(\sigma^k_{n_i,tree}, g) = ê(h_i, y_i) \times \prod_{x=1}^{x \le l} \sigma^k_{n_{child-x}}$
    EndIF
End Loop
```



$\sigma^k_{n_1,tree} = \sigma^k_{n_1}$
$\sigma^k_{n_2,tree} = \sigma^k_{n_2} \times \sigma^k_{n_1,tree}$
$\sigma^k_{n_5,tree} = \sigma^k_{n_5} \times \sigma^k_{n_2,tree}$
$\sigma^k_{n_7,tree} = \sigma^k_{n_7}$
S || other sign.
$n_2 \rightarrow n_5$
n_1 S $\rightarrow n_2$
n_1
n_2
n_5
n_7
S $n_7 \rightarrow n_c$
S $n_5 \rightarrow n_4$
n_4
$\sigma^k_{n_4,tree} = \sigma^k_{n_4} \times \sigma^k_{n_5,tree}$
$\sigma^k_{n_c,tree} = \sigma^k_{n_7,tree} \times \sigma^k_{n_8,tree}$
Collector
S $n_4 \rightarrow n_6$
n_3
n_6
$n_8 \rightarrow n_c$ S || other sign.
n_8
$n_3 \rightarrow n_6$ S
$n_6 \rightarrow n_8$ S || other sign.
$\sigma^k_{n_8,tree} = \sigma^k_{n_8} \times \sigma^k_{n_6,tree}$
$\sigma^k_{n_3,tree} = \sigma^k_{n_3}$
$\sigma^k_{n_6,tree} = \sigma^k_{n_6} \times \sigma^k_{n_4,tree} \times \sigma^k_{n_3,tree}$

Fig. 5. MST-based signature aggregation for smart meters in NAN

E. Integrated authentication solution

In this sub-section, a scenario is given as an example to demonstrate how our integrated solution executes: smart meter, n_X authenticates α blocks of packets, $\{B_1, B_2, \cdots B_\alpha\}$, and every block contains β packets, $\{M^{B_i}_1, \cdots M^{B_i}_\beta,\}$. First, *Alg.2-Signature Amortization* is used to sign every block in amortization. Eventually, α signatures $\{\sigma_1, \cdots \sigma_\alpha\}$ are generated for corresponding blocks. Each of them is encoded by IDA and sent to collector one by one. The collector decodes them by IDA but does not verify them individually. Instead, it uses *Alg.1-Batch Verification* to verify them in a bunch. Only one verification operation is required for all α blocks. During the data transportation in MST, *Alg.3-MST-based Signature Aggregation* is used by intermediate node (e.g. n_y) to aggregate signatures into single one which is sent to the collector. Failures occurred in *Alg.1* and *Alg.3* are diagnosed by trinary signature tree and MST-based verification tree, respectively.

V. Security Analysis, Performance Evaluation and Experimental Result

A. Security Analysis and Performance Evaluation

The security of our approach is based on digital signature schemes [9]. Signature aggregation, batch verification and signature amortization are proved to be secure in [29], [24], [25], respectively.

To deal with replay attacks, time-stamp can be added to packets so that the malicious attacks cannot reuse previous signatures. To protect this solution against Denial of Services (DoS) attacks, some solutions proposed in previous approaches could be borrowed: use distillation codes; then, forged packets and legal packets are separated into different categories; at last, erasing code is invoked over each category. Please refer to [26] for details.

TABLE II
Performance Evaluation

Schemes	Computation		Commun.
	collector	per/meter	
Per-Sign-Ver	$(\alpha\beta n)\boldsymbol{Ver}.$	$(\alpha\beta)\boldsymbol{Signing}$	$(c\alpha n)\ \lvert\boldsymbol{Sign}.\rvert$
Batch Verif.	$(\alpha n)\boldsymbol{Ver}$ $+ (\alpha\beta n)\boldsymbol{Mul}.$	$(\alpha\beta)\boldsymbol{Signing}$	$(c\alpha n)\ \lvert\boldsymbol{Sign}.\rvert.$
Sign. Aggre.	$(\alpha\beta n)\boldsymbol{Ver}.$	$(\alpha\beta)\boldsymbol{Signing}$ $+\boldsymbol{Mul}.$	$(\alpha n)\ \lvert\boldsymbol{Sign}.\rvert$
Sign. Amort.	$(\beta n)(\boldsymbol{Ver}.+\boldsymbol{Mul}.)$	$(\beta)\boldsymbol{Signing}$	$(c\alpha n)\ \lvert\boldsymbol{Sign}.\rvert$
Our proposal	$(n)\boldsymbol{Ver}.$ $+(\alpha\beta n)\boldsymbol{Mul}.$	$(\alpha)\boldsymbol{Signing}$ $+ (\alpha\beta)\boldsymbol{Mul}.$	$(\alpha n)\ \lvert\boldsymbol{Sign}.\rvert$

*c: constant; ***Mul.***: multiplication; $\lvert\boldsymbol{Sign}.\rvert$: bit length of signature;

Table II evaluates the communication and computation cost for scenario we mentioned in section IV (E). Notice that signature and verification operations are much heavier than multiplication [30]. We conclude that our proposal makes significant performance gains.

B. Experimental Test for Performance

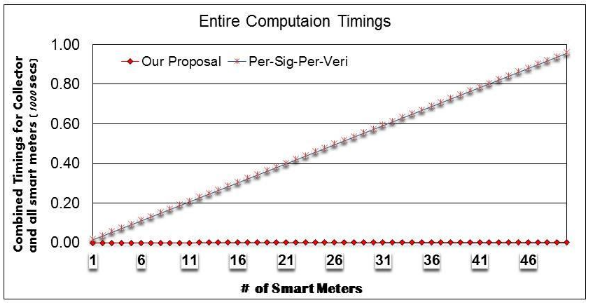


Fig. 6. Computational cost

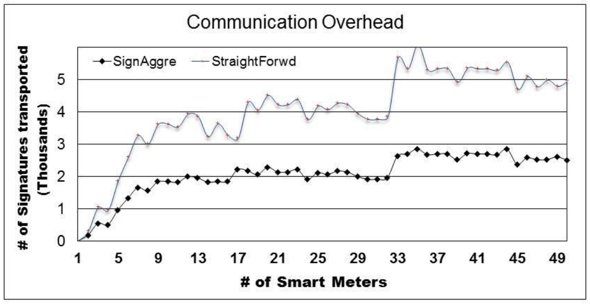


Fig. 7. Communication overhead

In [30], different signature schemes have been implemented on platform powered with PIII 1GHz CPU. The result shows that the timings to achieve a BLS signing and a BLS verification are 2.22ms and 45.8ms, respectively. The timings to implement a hash function and a scalar multiplication are less than 0.01ms. These experimental results will be utilized in our simulation.

Our proposal is simulated via Network Simulation-2 (ns-2) [35], a widely used simulation tool. This simulation utilized the test scenario: area (50 X 50 meters), 50 nodes (PIII 1GHz CPU for each node), 10 repetitions and mobility mode (10% mobile nodes). Fig. 6 and 7 demonstrate the communication and computational costs, respectively when 200 packets are sent by per smart meter (50 blocks and 4 packets per block). Fig. 6 shows that the entire computational timings are significantly decreased. It exactly matches with our performance assessment: the numbers of verification and signing operations are dropped by ${}^{1}/_{\alpha\beta}$ and ${}^{1}/_{\beta}$, respectively. Fig. 7 demonstrates that communication overhead is reduced by around 50% via deploying signature aggregation. Since signature amortization scheme reduces the number of signatures from β to 1 per block, our proposal reduces the overall communication cost in terms of authentication messages by $\approx {}^{1}/_{2\beta}$.

VI. Conclusion

Authentication scheme for data aggregation in smart grid system is a critical area for security research. Unfortunately, previous researches deploy the standardized authentication protocols or per-signing per-verification scheme to validate messages. They lack of performance optimization, be vulnerable to packet loss and cannot be resilient to DoS attacks. Furthermore, elementary tools to pinpoint the forged signature are not provided. In this paper, we integrated several efficient signature schemes to significantly reduce costs to achieve the authentication goal. Our proposal is an efficient scheme which is highly robust to signature packets loss. Most importantly, fault diagnosis algorithms are presented to detect failure points and minimize the fault execution times.

VII. References

[1] http://www.nist.gov/smartgrid/

[2] http://www.zigbee.org/

[3] *NIST Framework and Roadmap for Smart Grid Interoperability*. NIST Special Publication 1108. Release 1.0, Jan., 2010

[4] SG security working group (UCAIug) and NIST Cyber Security Coordination Task Group, "Security Profile for Advanced Metering Infrastructure," version 1.0, Dec. 10, 2009.

[5] *Guidelines for Smart Grid Cyber Security: Vol. 1, Smart Grid Cyber Security Strategy, Architecture, and High-level Requirements, NISTIR 7628*, NIST, 2010.

[6] K. Yagnik, S. Vadhva, R. Tatro and M. Vaziri, "California Smart Grid Attributes: California Public Utility Commission Metrics," *IEEE-Green*, pp. 1-6, April 2011.

[7] T. Flick and J. Morehouse, *Securing the Smart Grid: Next Generation Power Grid Security*, Elsevier Inc., 2010.

[8] AS. K.Pathan, *Security of Self-Organizing Networks: MANET, WSN, WMN, VANET*, CRC Press 2011.

[9] A. Menezes, PC Van Oorschot, *Handbook of applied cryptography*, CRC Press 1997.

[10] P. Mcdaniel and S. Mclaughlin, "Security and Privacy Challenges in the Smart Grid," *IEEE Security & Privacy*, vol. 7 no. 3, pp. 75-77, May/June 2009.

[11] D. Wei, Y. Lu, M. Jafari, P. Skare, and K. Rohde, "An Integrated Security System of Protecting Smart Grid against Cyber Attacks," in *Proc.* 2010 *IEEE PES Conf. on Innovative Smart Grid Technologies (ISGT)*, pp, 1-7.

[12] C. Bennett and S. B. Wicker, "Decreased Time Delay and Security Enhancement Recommendations for AMI Smart Meter Networks", in *Proc. 2010 IEEE PES Conf. on Innovative Smart Grid Technologies (ISGT)*, pp, 1-7.

[13] A. R. Metke and R. L. Ekl, "Security Technology for Smart Grid Networks," *IEEE Trans. On Smart Grid*, vol. 1, no. 1, pp. 99 – 107, June 2010.

[14] V. C. Gungor, B. Lu, and G. P. Hancke, "Opportunities and Challenges of Wireless Sensor Networks in Smart Grid," *IEEE Tran. On Industrial Electronics*, vol 57, no 10, pp. 3557 – 3564, Oct. 2010.

[15] D. Wu and C. Zhou, "Fault-Tolerant and Scalable Key Management for Smart Grid", *IEEE Trans. On Smart Grid*, vol. 2, no. 2, pp. 375 – 381, June 2010.

[16] F. Li, B. Luo and P. Liu, "Secure Information Aggregation for Smart Grids Using Homomorphic Encryption," in *Proc. 2010 IEEE Conf. Smart Grid Communication*, pp, 327-332.

[17] A. Bartoli, J. Hernandez-Serrano, M. Soriano, M. Dohler, A. Kountouris and D. Barthel, "Secure Lossless Aggregation for Smart Grid M2M Networks," in *Proc. 2010 IEEE Conf. Smart Grid Communication*, pp, 333-338.

[18] J. Zhang and C. A. Gunter, "Application-Aware Secure Multicast for Power Grid Communication," in *Proc. 2010 IEEE Conf. Smart Grid Communication*, pp, 339-344.

[19] H. K.-H. So, S. H.M. Kwok, E. Y. Lam and K.-S. Lui "Zero-Configuration Identity-based Signcryption Scheme for Smart Grid," in *2010 Proc. IEEE Conf. Smart Grid Communication*, pp. 321-326.

[20] J. Cho, M. Chung, K. Choi, Y. Lee, and J. Moon, "Enhanced Security Protocols for EPC Global Gen2 on Smart Grid Network," *in Proc. 2010 Inter. Conf. Ubiquitous Info. Technologies and Applications*. pp. 1-5.

[21] T. Baumeister, "Literature Review on Smart Grid Cyber Security", *Technical Report*, University of Hawaii, 2010.

[22] H. Khurana, R. Bobba, T. Yardley, P. Agarwal and E. Heine, "Design Principles for Power Grid Cyber-Infrastructure Authentication Protocols," in *Proc. 2010 43rd Hawaii International Conference on System Sciences*, pp.1-10.

[23] Y. Jiang, M. Shi, X. Shen, and C. Lin, "A Tree-Based Signature Scheme for VANETs," in *Proc. 2008 IEEE GLOBECOM*, pp. 1-5.

[24] L. Harn, "Batch Verifying Multiple RSA Digital Signatures," *IEE Electronic Letters*, vol 34, no. 12, pp. 1219-1220, June 1998.

[25] J. M. Park, E. K. P. Park, and H. J. Siegel, "Efficient Multicast Stream Authentication Using Erasure Codes," *ACM Trans. On Information and System Security*, vol 6, no.2, pp. 258-285, May 2003.

[26] C. Karlof, N. Sastry, Y. Li, A. Perrig, and J.D. Tygar, "Distillation Codes and Applications to DoS Resistant Multicast Authentication," *in Proc. 2004 Ann. Network and Distribut. Sys. Security Symp.*, pp. 37-56.

[27] D. Boneh, B. Lynn, and H. Shacham, "Short Signatures from the Weil pairing," *Journal of Cryptology*, vol 17. No. 4, pp.297-319, 2004.

[28] D. Boneh and M. Franklin, "Identity-based encryption from Weil pairing," in *Proc. of Crypto, LNCS 2001*, vol. 2139, pp. 213-229.

[29] D. Boneh, C. Gentry, B. Lynn and H. Shacham, "Aggregate and verifiably encrypted signatures from bilinear maps," *Advances in Cryptology, EUROCRYPT 2003*, LNCS, Volume 2656. pp. 416-432.

[30] P. Barreto, B. Lynn and M. Scott, "Efficient implementations for Pairing-based Cryptography," *Journal of Cryptology*, vol 17, no. 4, pp.321-334, 2004.

[31] C. Schwingenschlőgl and S. Eichler, "Certificate-based key management for secure communications in ad hoc networks," *In Pro. 2004 the 5th European Wireless Conference: Mobile and Wireless Systems beyond 3G*, pp. 498-504, Feb. 2004.

[32] R. E. Tarjan, "Sensitivity Analysis of Minimum Spanning Trees and Shortest Path Trees," *Info. Proc. Lett*. vol. 14, no. 1, pp. 30-33. 1982.

[33] D. Li and S. Sampalli, "Further Improvement and Vulnerability for Fast Encryption for Multimedia," *International Journal of Network Security*, vol.7, no.2, pp.188-193, Sep. 2008.

[34] D. Li and S. Sampalli, "A Hybrid Group Key Management Protocol for Reliable and Authenticated Rekeying," *International Journal of Network Security*, vol. 6, no. 3, pp. 270-281, 2008.

[35] http://www.isi.edu/nsnam/ns

[36] *Specification for the Advanced Encryption Standard (AES)*. Federal Information Processing Standards Publication 197, 2001.