Hindawi Publishing Corporation
International Journal of Digital Multimedia Broadcasting
Volume 2011, Article ID 382147, 9 pages
doi:10.1155/2011/382147

Research Article

BVS: A Lightweight Forward and Backward Secure Scheme for PMU Communications in Smart Grid

Wei Ren,[1] Jun Song,[1] Min Lei,[2] and Yi Ren[3]

[1] *School of Computer Science, China University of Geosciences, Wuhan 430074, China*
[2] *School of Software Engineering, Key Laboratory of Network and Information Attack and Defense Technology of MoE, Beijing 100876, China*
[3] *Department of Information and Communication Technology, University of Agder (UiA), Grimstad, Norway*

Correspondence should be addressed to Wei Ren, weirencs@gmail.com

Received 30 November 2010; Accepted 20 April 2011

Academic Editor: Pierangela Samarati

In smart grid, phaser measurement units (PMUs) can upload readings to utility centers via supervisory control and data acquisition (SCADA) or energy management system (EMS) to enable intelligent controlling and scheduling. It is critical to maintain the secrecy of readings so as to protect customers' privacy, together with integrity and source authentication for the reliability and stability of power scheduling. In particular, appealing security scheme needs to perform well in PMUs that usually have computational resource constraints, thus designed security protocols have to remain lightweight in terms of computation and storage. In this paper, we propose a family of schemes to solve this problem. They are public key based scheme (PKS), password based scheme (PWS) and billed value-based scheme (BVS). BVS can achieve forward and backward security and only relies on hash functions. Security analysis justifies that the proposed schemes, especially BVS, can attain the security goals with low computation and storage cost.

1. Introduction

Smart grid is envisioned as a long-term strategy for national energy independence, controlling emission, and combating global warming [1]. Smart grid technologies utilize intelligent transmission to deliver electricity, together with distribution networks to enable two-way communications. These approaches aim to improve reliability and efficiency of the electric system via gathering consumption data, delivering dynamic optimization of operations, and arranging energy saving schedules.

The smart grid promises to transform traditional centralized, producer-controlled network to a decentralized, consumer-interactive network. For example, consumers react to pricing signals delivered by control unit from smart meters to achieve active load adjustment. Supervisory control And data acquisition (SCADA) or energy management system (EMS) may collect one data points every 1 to 2 seconds, whereas phaser measurement units (PMUs) may collect 30 to 60 data points per second [2].

The security of smart grid is a critical issue for its applicability, development and deployment [3–7]. On one hand, the security, and especially the availability of power supplying system, affects homeland security, as it is an indispensable infrastructure for pubic living system [8–10]. That is, any transient interruption will result in economic and social disaster. On the other hand, introduction of end devices such as PMUs requests for data and communication security to support secure and reliable uploading of measurements [11, 12].

As the PMUs are exposed far from the central control unit, they present as a security boundary line between defenses and attacks. Such frontier may be tampered by curious users who intend to make certain profits or, even worse, hacked by malicious attackers who target for damaging power scheduling performance [13, 14]. For example, in the former case, advanced customers may try to reduce the value of meter's readings by revising circuits or interfering signals outside; curious eavesdroppers may be interested in customers' power-consuming patterns to pry about the consumers' privacy such as daily behaviors or schedules. In

the latter case, the attackers may invoke long-lasting peak values in meters to disturb the SCADA's scheduling strategies [15] or inject a worm to infect all meters to threaten the entire system resulting in a so-called billion-dollar bug [1]. To thwart the former security threaten, a straightforward way is to protect data confidentiality in PMU communications. Furthermore, to guarantee data integrity and data source authentication in PMU communications can mitigate the latter threat. Thus a prerequisite requirement arises—how to deliver and protect the encryption key and integrity key for such communications.

As the security issue for smart grid is an emerging new topic, currently available and customized solutions are a very few or undergoing development. Most existing work focuses on formulating the security problems [1–4, 7, 16] or build the security frameworks [5, 6, 8, 11, 17–19]. Especially, no existing work addresses key management issue for PMU communications in depth to the best of our knowledge. In this paper, we address data security and communication security between PMUs and control units in smart grid from the viewpoint of key management, including key generation, key deployment, and key evolution. After analyzing the data and communication security requirements, firstly, we propose several customized approaches and solutions.

Smart grid has some characteristics of itself own, for example, large number of deployment of end devices, real-time communications, resource constraints in tangible devices, to name a few. Thus, proper security solutions should beware of those specialities, for example, avoid disadvantages in applicable context to improve the applicability of proposed scheme. On the other hand, security design should be crafted for taking advantages of some properties of smart grid such as network architecture, domain context, and operational flow, as it may help to improve the overall performance. More specifically, we will dissect smart grid architecture to extract networking and system model and explore the inner mechanisms such as operational flow between SCADA and PMUs. We propose to solve the security problems by incorporating security scheme into the operational flow seamlessly to shrink the overall cost, and thus improve integrated performance. Such a design rationale makes use of inherent information in operational procedures as a security gradient and will be explained in the paper.

The major contributions of this paper are listed as follows.

(1) The security analysis, including network and system model, attack model and security requirements between PMU and SCADA communications in smart grid are extensively explored.

(2) We propose several lightweight schemes, including public key based scheme (PKS), password based Scheme (PWS), and billed value-based scheme (BVS) to tackle different application scenarios.

The rest of the paper is organized as follows. In Section 2, we discuss basic assumption and models used throughout the paper. Section 3 provides detailed description of our proposed schemes. We analyze security and performance aspects of the proposed scheme in Section 4. Finally, Section 5 concludes the paper.

2. Problem Formation

2.1. Network Model and System Model. The following related entities usually exist in smart grid applications.

(1) *PMU*s. Phaser measurement unit is an indispensable device for smart grid. As it may be exposed outside to potential adversaries, its security should be addressed firstly in the exploration of solutions. Each PMU has an automatic meter reading (AMR) to provide power values (or additionally a unit to delivery pricing information). The PMUs rely on the communication network to send measurements (or receive control instructions).

The characteristics of PMUs are as follows.

(i) Scalability. The number of PMUs is very large, could be in a scale of more than fifty thousands, depending on family quantity in a management domain, for example, in a county or a city scale.

(ii) Resource constraints. The computation and storage resources of PMUs are usually assumed to have constraints.

(iii) Compatibility. PMUs may have multiple variants for different categories of customers, and certain legacy systems or devices may try to be migrated or upgraded to a new version at first as more as possible.

(iv) Asymmetry. PMUs has a large volume of uploading traffics, but down-link traffics are comparatively much less than uploading traffics.

Therefore, incorporated security enhancement modules in PMUs should be affordable and have minimized revision for end customers. The design also needs to be suitable to multiple variants of PMUs, being compatible to general situations or legacy systems. We, thus, make the least assumption on the processing ability of PMUs. That is, their computation ability may be as low as a chip in sensor node, and storage space may be in megabyte magnitude. Thus, our solution can be applied in most architectures and perform better once higher processing platforms are available.

(2) *SGCC*. The control instructions are sent from smart grid control center, denoted as SGCC in this paper. It is a part of control unit of SCADA (or EMS). SGCC usually has enough computation and storage resources.

(3) *Customer*. The customers are always presented or related in smart grid applications; the PMU is not an isolated computing unit (this point is different with sensor node). Thus in security design, the human element could be considered and exploited if needed.

The communication network could be currently available home networks; we do not specify networking settings

such as topology and parameters such as bandwidth to make our discussion be suitable in most general situations.

To summarize, we observe that in smart grid, two asymmetry or unbalance present: computation resources in PMUs and in SGCC; uploading and downloading communication volumes. They may affect some subtle tradeoff in design of security schemes. Moreover, human interaction may also be incorporated into the design.

2.2. Attack Model. Similar to the statement in related work [2], traditional communications involve devices that are in areas with physical access controls (such as fences and locked houses), but smart PMUs are deployed in the areas that could be accessible by both consumers and adversaries. Consequently, we have to assume that PMUs are located in a hostile environment and much stronger adversaries exist.

As attackers are assumed to be malicious and intended to tamper the system to gain some profits. For example, attackers may reduce the transmission data such as power consuming value for lower payment; they may also pulse the data in the transmission to corrupt the scheduling strategies; they may pry the communication patterns to speculate the privacy of consumers, such as daily behaviors; they may inject any forgeable data such as consuming value into the communication by manipulating PMUs, or even aim to crash partial or entire SCADA or EMS.

2.3. Security Requirement. The data are generated from PMUs and transmitted into communication networks. To protect such kind of data, security scheme should the fulfill security requirements as follows:

Data Confidentiality. Data confidentiality in transmission should be protected. Otherwise, utility consumption values will be known by attackers, which will leak much information on consumers' behaviors.

Data Integrity. Integrity of the data transferred in communication should be guaranteed so that any modification of the data can be detected.

Data Source Authentication. The source of the data should be verifiable by receivers to confirm the data authenticity and so as to exclude forged data.

Above requirements should be addressed in context of smart grid, or the solution should concern its applicability in smart grid environments. Such security requirements inevitably ask for a prerequisite demand—key management issue. The related keys such as the encryption key, integrity key, or authenticity key can be available and properly protected.

2.4. Design Goal. Based on above observations, we state the design goal as follows:

We search for a highly robust but lightweight scheme to protect data confidentiality, data integrity, and data source authentication in case of the presence of strong attackers and in the context of smart grid. To fulfill such objectives, the prerequisite is how to generate, manage, and refresh underlying keys such as encryption key, integrity key, and authentication key. The encryption key is called session encryption key, denoted as SEK, which is only used for one uploading session of power values. The authentication key and integrity key are combined (or interchangeable) together and is called session integrity key, denoted as SIK, which is also used only for one session. The session period depends on the gathering interval of power values, scheduling policy, and security strength.

Table 1: Notation.

PKS	Public Key based Scheme
MK	Master Key
UID	Unique ID
PMU	Phaser Measurement Units
SGCC	Smart Grid Control Center
SEK	Session Encryption Key
PubK	Public Key
PriK	Private Key
PWS	Password based Scheme
PWD	Password
BVS	Billed Value based Scheme
V	Value of automatic meter readings
ETK	Evolutionary Transportation Key
$\{M\}_K$	Encryption of M using key K
$\|\|$	Concatenation

3. Proposed Schemes

In this section, we investigate a family of schemes for better understanding and explaining motivations. Each latter scheme may improve previous one by addressing some of its limitations in terms of performance or usability, or deal with several subtle tradeoff to achieve better overall performance and security.

We list all major notations used in the remainder of the paper in Table 1.

3.1. Public Key based Scheme—PKS. We firstly propose a basic public key based scheme, called PKS, to illustrate our motivations. To facilitate the encryption and message authentication code, the encryption key and integrity key are required. The naive scheme is using predistributed master key (MK) in the PMUs, but this solution has one weakness—if a PMU is compromised, the MK in this PMU will be leaked. Therefore, all derived keys from MK will be exposed if such derivation is only related to MK.

Each PMU has a MK that is preloaded into the PMU upon the deployment. PMU always has a unique id, called UID, which could be a designated sequence number of the PMU upon deployment and stored in on-chip read-only memory. Similar to MK, the UID is also stored by SGCC after the deployment of PMU. Customers are assumed to have a certificated public key generated by certificate authority (CA), usually a trusted third party for smart grid.

The Public Key based scheme (PKS) is described as following stages:

(1) Stage I—Preparation. Before session key establishment, SGCC checks whether the customer's PubK is revoked from Revocation List (RL). If not, go to next stage. Otherwise, stop or choose another scheme.

(2) Stage II—Session key seed establishment.

(2.1) Establishment request (SKER). SGCC selects a random number $R = \{R_1 \| R_2\}$ and sends R that is encrypted by a customer's PubK to PMU. That is,

$$\text{SGCC} \longrightarrow \text{PMU} : \{Tm \| R_1 \| R_2\}_{\text{PubK}}, \quad (1)$$

where Tm is a time stamp denoting current time.

(2.2) Establishment acknowledgement (SKEA). The customer relies her private key (PriK) to decrypt out R, checks whether Tm is in the proper range and sends back as follows:

$$\text{PMU} \longrightarrow \text{SGCC} : \{R_1 + 1 \| R_2 - 1\}_{\text{PriK}}. \quad (2)$$

(3) Stage III—session key generation.

(3.1) SEK generation. The customer uses following method to generate session encryption key: $\text{SEK} = \text{Hash}(R_1 \| \text{UID} \| \text{MK})$.

(3.2) SIK generation. The customer uses following method to generate session integrity key: $\text{SIK} = \text{Hash}(R_2 \| \text{UID} \| \text{MK})$.

(4) Stage IV—data transmission. The PMU sends power value to SGCC:

$$\text{PMU} \longrightarrow \text{SGCC} : \{V\}_{\text{SEK}}, \{\text{Hash}(V)\}_{\text{SIK}}, \quad (3)$$

where V is a value of meter reading in the last sample period.

Remarks.

(1) Two random values (R_1 and R_2) are used instead of one random value, can protect SEK and SIK independently. The exposure of one random number (i.e., R_1 in SEK or R_2 in SIK) will not result in the leakage of the other.

(2) Tm is used for defending replay attack. If Tm is not in the range, the SKER (session key establishment request) message will be ignored. The replay attack may result in DoS (denial of service) attack to PMUs. It can be mitigated by security policy that is out of the scope of the paper.

(3) SKEA (session key establishment acknowledgement) message confirms the authenticity of PMU and synchronizes the generation of session keys.

(4) SEK generation relies on $\{R_1 \| \text{UID} \| \text{MK}\}$. If MK is compromised by software flaws in program, UID may remain secure due to its hardware compromising hardness (namely, tamper-proofed read-only memory accessed only by PMU). R_1 guarantees the freshness and authenticity of SEK.

(5) The confidentiality and integrity of power data V are guaranteed by SEK and SIK.

Security Analysis. The security of SEK and SIK is guaranteed by the security of PriK. If the PriK is safely possessed, attackers cannot recognize SEK and SIK. UID is used for generating SEK and SIK, which increases the difficulties of hardware tampering by attackers. That is, even if MK is leaked by software compromising, the UID may still sustain secrecy, because it is a hardware extracted value and not easy to be revealed by only software compromising. Moreover, even though the attacker can further reveal UID by hardware scrutiny physically, they cannot be able to possess the PriK simultaneously, as it is securely and personally held by customers. The customers are assumed to safely possess their PriK and only use such keys in session establishment stage. Therefore, the security of SEK and SIK can be guaranteed.

Performance Analysis. We mainly consider performance at PMU side. The session key seed establishment stage induces the operations are 1 public key decryption and 1 public key encryption.

The session key generation stage incurs 2 hash function computation. Data transmission stage has 2 symmetric key encryption and 1 hash function computation. The communication includes two messages for key establishment and one message for data transmission. It shows that the operations in this stage almost remain to being minimized.

Usability and Cost Analysis. PKS involves customers' PriK, so customers usually have to possess some local device such as a USB disk to store such key (that is distributed by third trusted party). It may induce a USB port in PMU attached devices, which increases the cost of PMU devices. It also demands customers to safely possess a portable USB key, which may add customers' burden. The RL (revocation list) must be maintained or synchronized with CA (certificate authority) by SGCC. Or, SGCC will retrieve the RL if RL is only maintained by CA, it will introduce some response delay due to RL retrieval.

The customers are asked for participating the key establishment stage only when session keys are generated or updated. In proposed scheme, the session key establishment (or updating) stage are launched by SGCC, if customers and SGCC previously agree on one or multiple time-slots, for example, each Sunday 10:00 PM, which is only a management issue. If needed, the session key updating request can also be launched by customers, in this case customers may communicate with SGCC offline to negotiate a proper time-slot.

3.2. Password based Scheme: PWS. The public Key based Scheme (PKS) scheme is resilient to software compromise, but it assumes the existence of PKI (Public Key Infrastructure) system. To avoid such a restricted assumption, we propose a password based scheme, called PWS, to improve the flexibility and usability of the proposed solution.

In this scheme, password (denoted as PWD) is induced, which is a easily memorized (at least 8) digits in range [0, 9] so that PKI becomes unnecessary. The password are usually selected by customers for their favors so as to be easily memorized. The password usually is uploaded to SGCC off-line, for example, when create or start up the utility account or upon the deployment of PMUs.

The PWS scheme is described as follows:

(1) Stage I—session key seed establishment.

(1.1) Establishment request (SKER). SGCC selects a random number $R = \{Tm \| R_1 \| R_2\}$ and sends R encrypted by the customer's password PWD to PMU. That is

$$\text{SGCC} \longrightarrow \text{PMU} : \{Tm \| R_1 \| R_2\}_{\text{PWD}}, \tag{4}$$

where Tm is current time stamp.

(1.2) Establishment acknowledgement (SKEA). The customer decrypts out R via her PWD, checks whether Tm is in the range, and sends back as follows:

$$\text{PMU} \longrightarrow \text{SGCC} : \{R_1 + 1 \| R_2 - 1\}_{\text{PWD}}. \tag{5}$$

(2) Stage II—session key generation.

(2.1) SEK generation. The customer uses following method to generate session encryption key:

$$\text{SEK} = \text{Hash}(R_1 \| \text{UID} \| \text{MK} \| \text{PWD}). \tag{6}$$

(2.2) SIK generation. The customer uses following method to generate session integrity key:

$$\text{SIK} = \text{Hash}(R_2 \| \text{UID} \| \text{MK} \| \text{PWD}). \tag{7}$$

(3) Stage III—data transmission. The customer sends power value to SGCC as follows:

$$\text{PMU} \longrightarrow \text{SGCC} : \{V\}_{\text{SEK}}, \{\text{Hash}(V)\}_{\text{SIK}}. \tag{8}$$

Enhancement. In PKS scheme, once PriK is exposed, all random number R are revealed. It is appealing that certain previous keys and together ciphertext encrypted by such keys are still safe even if current keys are exposed. This situation is so-called forward secrecy. To further enhance the security of scheme PWS, we propose to use key evolution method. Here, we call the key used for transporting random number R is a transportation key. We propose to use one-time hash value of PWD as the transportation key, and use hash chain as a key evolution strategy.

More specifically, at ith time in Stage I the encrypted key is not PWD but $\text{Hash}^{(i)}(\text{PWD})$, $(i \geqslant 1)$. That is, assuming at the ith time (or session) of transmission of random number R. The two steps in stage I are revised as follows.

(1.1) SGCC selects a random number $R = \{R_1 \| R_2\}$ and sends R to PMU that is encrypted by hashed customer's password. That is,

$$\text{SGCC} \longrightarrow \text{PMU} : \{Tm \| R_1 \| R_2\}_{\text{Hash}^{(i)}(\text{PWD})}. \tag{9}$$

(1.2) The customer uses $\text{Hash}^{(i)}(\text{PWD})$ to decrypt out R, checks whether Tm is in the range, and sends back:

$$\text{PMU} \longrightarrow \text{SGCC} : \{R_1 + 1 \| R_2 - 1\}_{\text{Hash}^{(i)}(\text{PWD})}. \tag{10}$$

In this way, the encryption key for random number transportation will be evolved very time and be used for only once. Even attackers can reveal one encryption key, for example, $\text{Hash}^{(i)}(\text{PWD})$, they cannot conjecture previous encryption keys such as $\text{Hash}^{(j)}$ (PWD) $(1 \leqslant j < i)$. The reason comes from the one-wayness of hash function. That is, given the image of the function, it is computationally infeasible to compute preimage. Therefore, this enhancement guarantees the forward secrecy of the transportation key.

In addition to the introduction of key evolution, a value SALT is further suggested to strengthen the limited length of PWD and defend off-line dictionary attack. As PWD needs to be easily memorized, the length of PWD usually is no longer than 8 digits. To extend the off-line brute force search space, SALT value can be used. The SALT will be safely stored in SGCC and PMU, respectively. The indeed PWD used for key evolution will be {PWD‖SALT} instead of PWD.

Security Analysis. The security of SEK and SIK is guaranteed by the security of PWD. SGCC is always assumed to securely possess the PWD, as it is in a trusted domain. If PWD is only memorized by consumers and its secrecy is maintained when it is typed on keyboard, attackers cannot recognize SEK and SIK due to the unknownness of PWD. Without PWD, attackers cannot reveal R that is the generating ingredient of session keys (namely, SEK and SIK).

Moreover, UID and MK are incorporated in the generation of SEK and SIK for the similar reason with public key based scheme. Especially, PWD is also proposed to embed into the generation of SEK and SIK, to further enhance the session key's secrecy.

The scheme can further provide forward secrecy of transportation key used for transferring random number R. That is, even a key $\text{Hash}^{(i)}$ (PWD) for encryption of R are exposed, the keys such as $\text{Hash}^{(j)}$ (PWD) $(1 \leqslant j < i)$ still remain secret.

Based on above analysis, the security of SEK and SIK can be guaranteed.

Performance Analysis. The operations induced at PMU side are 1 symmetric key decryption, 1 symmetric key encryption for key establishment, and 2 hash function computation for key generation.

Usability and Cost Analysis. PWS scheme may ask users to plug a small numeric keyboard into PMU attached device

or PMU itself incorporates such a numeric keyboard panel, which slightly increases the cost of PMU. Nonetheless, the usability of scheme PWS is better than scheme PKS, as the USB key is not required and numeric keyboard is much cheaper than USB key. The PWD can be reinstated or updated by customers via off-line channel with SGCC.

3.3. Billed Value-Based Scheme: BVS. PWS scheme can further be improved to avoid typing password by customers. We further propose a more lightweight scheme by using billed value. The billed value here means the last billed value for consumed utility. We assume that value is only known by SGCC and PMU, as we assume the utility consumption result is a private information of customers and should be kept secret (that is the underlying goal of proposed schemes). Suppose the billed value is BV, we use Hash(BV) to replace the functionality of PWD in PWS scheme. Note that the BV is always not equal to real-time consuming value V, as the billing period is always longer than data gathering period and billed value is a constant value in last billing period. For example, BV is altered once in one billing period, but the gathering value V is collected by SGCC much more frequently (depending on the control and scheduling strategies). The enhancement rationale for forward secrecy in PWS scheme can also be migrated to BVS scheme. Thus, the proposed BVS scheme is as follows.

(1) Stage I—session key seed establishment.

(1.1) SGCC selects a random number $r = \{R_1 \| R_2\}$ and sends R encrypted by Hash(BV) to PMU. That is,

$$\text{SGCC} \longrightarrow \text{PMU} : \{R_1 \| R_2\}_{\text{Hash}^{(i)}(\text{Hash}(\text{BV}))}. \tag{11}$$

(1.2) Customers use their $\text{Hash}^{(i)}(\text{Hash}(\text{BV}))$ to decrypt out R.

(2) Stage II—session key generation.

(2.1) SEK generation. The customer uses following method to generate session encryption key:

$$\text{SEK} = \text{Hash}(R_1 \| \text{UID} \| \text{MK} \| \text{BV}). \tag{12}$$

(2.1) SIK generation. The customer uses following method to generate session integrity key:

$$\text{SIK} = \text{Hash}(R_2 \| \text{UID} \| \text{MK} \| \text{BV}). \tag{13}$$

(3) Stage III—data transmission. The customer sends power value to SGCC as follows:

$$\text{PMU} \longrightarrow \text{SGCC} : \{V\}_{\text{SEK}}, \{\text{Hash}(V)\}_{\text{SIK}}. \tag{14}$$

Enhancement.

(1) The hash functions used in our scheme could be the same function or different functions, depending on the available storage space for implementation of hash function code. We suggest to use different hash functions as it will be more secure. That is, the transportation key is $\text{Hash}_1^{(i)}(\text{Hash}_2(\text{BV}))$.

(2) The synchronization of SGCC and PMU on BV is straightforward if the clock of PMU and SGCC can be strictly synchronized. Normally, the transaction is cutoff at the end of billing period, for example, at the 1:00 AM of the first day of each month. At that moment, the PMU will save this value (also the last uploading value) before 1:00 AM as a billing value. That value is the utility consumption of last payment period. Hence, it performs as a common shared secret between PMU (customer) and SGCC in a natural way, and it is automatically and periodically updated to maintain freshness.

(3) If the clock of PMU is not strictly synchronized, we propose the following policy—multiple transmission of BV. That is, in the first day of each month before 1:00 AM, PMU stores any current V as BV and attaches it to multiple messages. That is,

$$\begin{aligned} \text{PMU} \longrightarrow \text{SGCC} : &\{V \| \text{BV} \| \text{BV}_{\text{TAG}}\}_{\text{SEK}}, \\ &\{\text{Hash}(V \| \text{BV} \| \text{BV}_{\text{TAG}})\}_{\text{SIK}}, \end{aligned} \tag{15}$$

where BV_{TAG} presents a tag for notifying SGCC that the BV is attached.

(4) As the success of update of BV is critical for key synchronization, we propose another policy by using confirmed reply if two-way communication is available (in fact, two-way communication is usually available in smart grid). That is, the two-way messages include confirmation from SGCC on the receipt of BV, as follows:

$$\begin{aligned} \text{PMU} \longrightarrow \text{SGCC} : &\{V \| \text{BV} \| \text{BV}_{\text{TAG}}\}_{\text{SEK}}, \\ &\{\text{Hash}(V \| \text{BV} \| \text{BV}_{\text{TAG}})\}_{\text{SIK}}, \\ \text{SGCC} \longrightarrow \text{PMU} : &\{\text{BV} \| \text{BV}_{\text{TAGACK}}\}_{\text{SEK}}, \\ &\{\text{Hash}(\text{BV} \| \text{BV}_{\text{TAGACK}})\}_{\text{SIK}}. \end{aligned} \tag{16}$$

(5) In PWS scheme, only forward secrecy of the transportation key are ensured. That is, from $\text{Hash}_2^{(i)}(\text{BV})$ attackers can compute $\text{Hash}_2^{(i+1)}(\text{BV})$, but not $\text{Hash}_2^{(i-1)}(\text{BV})$. To further enhance the secrecy of transportation key, we propose an enhancement method for both forward secrecy and backward secrecy. Here, backward secrecy means that even if the current key is exposed, the future keys cannot be correctly conjectured. Concretely, we propose the

following key evolution strategy, assuming the ith transportation of random number R:

$$\begin{aligned} \text{SEED} &\longleftarrow \text{Hash}_2(BV), \\ \{L\|R\} &\longleftarrow \text{SEED}, \\ \text{ETK}_L &\longleftarrow \text{Hash}^{(n+1-i)}(L), \\ \text{ETK}_R &\longleftarrow \text{Hash}^{(i)}(R), \\ \text{ETK} &\longleftarrow \{\text{ETK}_L\|\text{ETK}_R\}, \\ \text{SGCC} &\longrightarrow \text{PMU} : \{R_1\|R_2\}_{\text{ETK}}, \end{aligned} \tag{17}$$

where ETK means evolutionary transportation key, used for the transportation of random number R.

In this way, the encryption key for random number will be changed very time and be used for only once. Even attackers can reveal one encryption key, for example, $\text{Hash}^{(n+1-i)}(L)$, they cannot conjecture future encryption keys such as $\text{Hash}^{(n+1-j)}(L)(i < j < n)$. The reason is the one-wayness of hash function. Therefore, the backward secrecy of transportation key ETK is guaranteed. Besides, the value of n is stored in the table of database in SGCC and preloaded into PMU.

(6) As the key is updated in bidirections to provide forward and backward secrecy, the backward secrecy needs the predetermination of maximal evolution times $n + 1$. If the maximal number is reached, or if sampling times i is increased to $n + 1$ within one billing period, we propose to update BV by using $\text{BV} \leftarrow \text{Hash}_2(\text{BV})$ and reset i to 1. The updating period of transporting key result in the alternation of random number R, and further SEK and SIK. The updating period depends on the security policy and data gathering frequency.

(7) In one billing period, all SEKs and SIKs are generated by different random number R that are transported using the derived keys from same BV. As key evolution is involved, the loss of synchronization of i at PMU and SGCC in one billing period will result in the decryption failure of random number R at the PMU side. The minor adjustment can solve this problem. At SGCC side, the random number could have some relation between R_1 and R_2, for example, $R = \{R_1\|R_2 = \text{Hash}_1(R_1)\}$. At PMU side upon receipt of $\{R_1\|R_2\}_{\text{ETK}}$, PMU will decrypt it with supposed ETK. If decrypted value presents the designated relation, the synchronization is maintained.

(8) As the security of BV is critical for BVS scheme, we also propose some strategies to protect BV's secrecy. The secret BV can be a function of public BV. We leave some flexibility of BV value's customized tuning. For example, customers can select a policy on how to generate secrete BV from public BV value, when paying for the utility, and upload the selection into SCADA. Such policy could be an option in policy list. The synchronization of BV between PMU and SCADA can be confirmed by customers upon paying for the utility bill and checking PMU. We assume PMU has a screen to display assumed BV when customers type designated on-board button, and customer can also upload PMU her selected policy by pressing the same button. The public BV will not lead to the exposure of secrete BV, unless corresponding policy is exposed.

Security Analysis. The basic analysis is similar to PWS scheme. The security of SEK and SIK is guaranteed by the security of BV, which is assumed to be the private information of customers. The one-time usage of BV derived encryption key improve the confidentiality of random number, which is an ingredient of SEK and SIK. Together with UID, MK and BV, the security of SEK and SIK can be guaranteed.

Especially, the bidirection hash-chain based derivation guarantees both forward and backward secrecy of the transportation key ETK, so both forward and backward secrecy of R are maintained. As R is the seed of session key SEK and SIK, the both forward and backward secrecy of session keys are guaranteed.

Performance Analysis. The scheme induces the operations are 1 symmetric key encryption, symmetric key decryption, 2 hash function computation. The SEK and SIK generation requires a one-way message from SGCC to PMU. Besides, the enhancement induces more computation, but all are hash function calculation that have low overhead.

Usability and Cost Analysis. Last available BV value can be stored in SGCC and PMU, so customers do not need to remember a password. The security of BV is critical, so it may save in some separated devices such as on-chip rewritable memory to protect its secrecy.

4. Analysis

4.1. Security Analysis. We state the analysis formally by presenting following propositions.

Proposition 1. *If PriK is secretly possessed, PKS will be secure.*

Proof. If PriK is secretly possessed, R will remain secure. If R is secret, computation of SEK and SIK is equivalent to random guess due to the one-wayness of hash function. Thus, if SEK is a secret, the data secrecy of V will be ensured. If SIK is a secret, the message integrity and message source authentication will be ensured. The reason is the one wayness of hash function and the secrecy of SIK. □

Proposition 2. *If PWD is maintained secret, the scheme PWS will achieve security goals.*

Proof. Straightforward. □

Definition 3. Forward secrecy. Given a key K, it is computationally infeasible to conjecture K_f, where K_f is the key

before last key evolution. That is, $|\Pr\{K_f \mid K, K \leftarrow f(K_f)\} - \Pr\{K_f\}| < \epsilon(n)$, where $\epsilon(n)$ is a negligible polynomial related to a security parameter n. n usually is the security strength in length. $f(\cdot)$ is the key evolution function. $\Pr\{K_f\}$ denotes the probability of revealing K_f.

Definition 4. Backward secrecy. Given a key K, it is computationally infeasible to conjecture K_b, where K_b is the key after key evolution. That is, $|\Pr\{K_b \mid K, K_b \leftarrow f(K)\} - \Pr\{K_b\}| < \epsilon(n)$, where $\epsilon(n)$ is a negligible polynomial related to security parameter n. n usually is the security strength in length. $f(\cdot)$ is the key evolution function. $\Pr\{K_b\}$ denotes the probability of revealing K_b.

Lemma 5. *One-way function is sufficient for forward secrecy, and necessary for forward secrecy if only one evolutionary key is stored and shared by communication peers.*

Proof. The proof for sufficient condition is straightforward. Next, we proof it is a necessary condition. As only one evolutionary key is stored and shared at communication peers, denoted as K, the next generated key is the function of K. That is, $K_f = f(K)$, where $f(\cdot)$ is a function taking as input K. If $|\Pr\{K_f \mid K\} - \Pr\{K_f\}| = 0$, we have $\Pr\{f(K) \mid K\} = \Pr\{f(K)\}$. $f(\cdot)$ is thus a real random number generation, for example, randomly sample function from a key space. In other words, $f(K) = K_f, (K_f \leftarrow_r \{0,1\}^{|K|})$, where $\leftarrow_r$ means random selection. As communication peers use key evolution for secure communication, they have to maintain holding a shared secret after key evolution. Thus, if $f(\cdot)$ is a real random number generation (RNG), the key evolution is unserviceable because the shared pairwise secret is lost. To maintain holding shared secret after key evolution, it needs to satisfy $|\Pr\{K_f|K\} - \Pr\{K_f\}| < \epsilon(n)$, where $\epsilon(n)$ is a negligible polynomial related to security parameter $n = |K|$. Therefore, $f(\cdot)$ has to take as input K and must has one-wayness, as desired. □

Proposition 6. *BVS has optimal forward and backward secrecy.*

Proof. For forward and backward security, one-way function is required. BVS guarantees forward and backward security by using only single one-way function. The first half key is generated by forward secure key evolution method; the other half key is generated by backward secure key evolution method. Due to the one-wayness of one-way function, the reveal of key after key evolution can only be done by random guess. Suppose $\text{ETK}_L = L_1, \text{ETK}_R = L_2$. That is, $\Pr\{K_b \mid K, K_b \leftarrow f(K)\} = 1/2^{L_1}$, and $\Pr\{K_f \mid K, K \leftarrow f(K_f)\} = 1/2^{L_2}$. Besides, $L_1 + L_2 = |K| = n$. Thus, $\Pr\{K_f, K_b \mid K, K \leftarrow f(K_f), K_b \leftarrow f(K)\} = 1/2^{L_1} \times 1/2^{L_2} = 1/2^{L_1+L_2} = 1/2^n$. Thus BVS has forward and backward secrecy.

Next, we proof it is optimal. If we define overall security strength is the minimum of backward secrecy strength and forward secrecy strength, the key revealing probability will be $\text{MAX}\{1/2^{L_1}, 1/2^{L_2}\}$. Thus, when $L_1 = L_2 = |K|/2$, the overall secrecy strength achieves an optimal strength $1/2^{n/2}$. □

Table 2: Scheme comparison.

Scheme	Security	Performance (PMU side)	Usability
PKS	PriK	PKD + Hash	PKI + USB Key
PWS	PWD + forward security	SKD + Hash	Keyboard
BVS	BV + forward + backward security	SKD + Hash	/

Above proof is valid even for any attack model for communication link (of course, we only consider any attack models that have finite computational ability, namely, polynomial attackers).

Proposition 7. *Scheme PKS is not forward secure, but scheme PWS is forward secure.*

Proof. Straightforward. □

Proposition 8. *Scheme PWS is not backward secure, but scheme BVS is forward and backward secure.*

Proof. In BVS scheme, $\text{ETK}_L \leftarrow \text{Hash}^{(n+1-i)}(L)$. On one hand, if ETK_L is exposed, attacker cannot conjecture future encryption keys $\text{Hash}^{(n+1-j)}(L)(i < j < n)$. Thus, the backward secrecy are guaranteed. On the other hand, if $\text{ETK}_R \leftarrow \text{Hash}^{i}(R)$ is exposed, attacker cannot conjecture future encryption keys $\text{Hash}^{(j)}(R)(j < i)$. Thus, the forward secrecy is ensured. Hence, either ETK_L or ETK_R cannot be conjectured. ETK, thus, has forward secrecy and backward secrecy, as desired. □

4.2. Performance Analysis. The additional computation of BVS only involves hash functions, so the computational cost is manageable. The hash function codes can be reused. For example, one hash function is SHA256; the other is SHA512. The incurred storage for codes is also lightweight. As hash function is typical lightweight cryptographic primitives, it is also extensively applied in computing platforms with much lower computational ability than PMU such as RFID tag [20]. Moreover, the hardware implementation of hash functions has competitive performances [21–23], which further guarantee the applicability of hash functions in PMUs.

Regarding the usability, BVS has the best performances. It has no requirement for PKI comparing with PKS scheme and has no requirement for password inputting device comparing with PWS scheme. We list the comparisons between three schemes in Table 2. (Acronym: PKD—public key decryption and SKD—symmetric key decryption.)

5. Conclusion

In this paper, we proposed a family of lightweight security schemes for session key seed establishment and session key

generation to guarantee data secrecy, data integrity, and data source authentication in communications from PMUs to SGCC (SCADA or EMS control center). We proposed public key based scheme (PKS) and password based scheme (PWS) for different application scenarios. Billed value-based scheme (BVS) was proposed and emphasized, as it can achieve forward and backward security by only relying on hash functions and has appealing usability or flexibility. Security and performance analysis justified that the proposed scheme BVS can achieve forward and backward secrecy with lightweight hash function computation.

Acknowledgments

This work was supported by Special Fund for Basic Scientific Research of Central Colleges, China University of Geosciences (Wuhan) under Grant no. 090109, Major State Basic Research Development Program of China (973 Program) (no. 2007CB311203), National Natural Science Foundation of China (no. 60821001), and Ph.D. Programs Foundation of Ministry of Education of China (no. 20070013007).

References

[1] P. McDaniel and S. McLaughlin, "Security and privacy challenges in the smart grid," *IEEE Security and Privacy*, vol. 7, no. 3, pp. 75–77, 2009.

[2] H. Khurana, M. Hadley, N. Lu et al., "Smart-grid security issues," *IEEE Security and Privacy*, vol. 8, no. 1, pp. 81–85, 2010.

[3] F. Boroomand, A. Fereidunian, M. A. Zamani et al., "Cyber security for smart grid: a human-automation interaction framework," in *Proceedings of the IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT Europe '10)*, pp. 1–6, November 2010.

[4] S. Clements and H. Kirkham, "Cyber-security considerations for the smart grid," in *Proceedings of the 2010 IEEE Power and Energy Society General Meeting (PES '10)*, pp. 1–5, September 2010.

[5] A. R. Metke and R. L. Ekl, "Security technology for smart grid networks," *IEEE Transactions on Smart Grid*, vol. 1, no. 1, pp. 99–107, 2010.

[6] D. Wei, Y. Lu, M. Jafari et al., "An integrated security system of protecting smart grid against cyber attacks," in *Proceedings of the Innovative Smart Grid Technologies (ISGT '10)*, pp. 1–7, 2010.

[7] M. Amin, "Challenges in reliability, security, efficiency, and resilience of energy infrastructure: toward smart self-healing electric power grid," in *Proceedings of the IEEE Power and Energy Society General Meeting (PES '08)*, pp. 1–5, Pittsburgh, Pa, USA, July 2008.

[8] G. N. Ericsson, "Cyber security and power system communication—essential parts of a smart grid infrastructure," *IEEE Transactions on Power Delivery*, vol. 25, no. 3, pp. 1501–1507, 2010.

[9] J. T. Seo and C. Lee, "The green defenders," *IEEE Power and Energy Magazine*, vol. 9, no. 1, pp. 82–90, 2011.

[10] J. Kim and J. Lee, "A model of stability," *IEEE Power and Energy Magazine*, vol. 9, no. 1, pp. 75–81, 2011.

[11] K. M. Rogers, R. Klump, H. Khurana, A. A. Aquino-Lugo, and T. J. Overbye, "An authenticated control framework for distributed voltage support on the smart grid," *IEEE Transactions on Smart Grid*, vol. 1, no. 1, pp. 40–47, 2010.

[12] Y. Wang, I. R. Pordanjani, and W. Xu., "An event-driven demand response scheme for power system security enhancement," *IEEE Transactions on Smart Grid*, vol. 2, no. 1, pp. 23–29, 2011.

[13] K. Moslehi and R. Kumar, "A reliability perspective of the smart grid," *IEEE Transactions on Smart Grid*, vol. 1, no. 1, pp. 57–64, 2010.

[14] Y. Wang, W. Li, and J. Lu, "Reliability analysis of wide-area measurement system," *IEEE Transactions on Power Delivery*, vol. 25, no. 3, pp. 1483–1491, 2010.

[15] J. Ma, P. Zhang, H. j. Fu et al., "Application of phasor measurement unit on locating disturbance source for low-frequency oscillation," *IEEE Transactions on Smart Grid*, vol. 1, no. 3, pp. 340–346, 2010.

[16] K. Budka, J. Deshpande, J. Hobby et al., "Geri—bell labs smart grid research focus: economic modeling, networking, and security amp; privacy," in *Proceedings of the 1st IEEE International Conference on Smart Grid Communications (SmartGridComm '10)*, pp. 208–213, November 2010.

[17] A. Vaccaro, M. Popov, D. Villacci, and V. Terzija, "An integrated framework for smart microgrids modeling, monitoring, control, communication, and verification," *Proceedings of the IEEE*, vol. 99, no. 1, pp. 119–132, 2011.

[18] T. Zhang, W. Lin, Y. Wang et al., "The design of information security protection framework to support smart grid," in *Proceedings of the 2010 International Conference on Power System Technology (POWERCON '10)*, pp. 1–5, 2010.

[19] T. M. Overman and R.W. Sackman, "High assurance smart grid: smart grid control systems communications architecture," in *Proceedings of the 1st IEEE International Conference on Smart Grid Communications (SmartGridComm '10)*, pp. 19–24, November 2010.

[20] T. L. Lim and Y. Li, "A security and performance evaluation of hash-based rfid protocols," in *Proceedings of the 5th China International Conferences on Information Security and Cryptology (Inscrypt '09)*, vol. 5487 of *Lecture Notes in Computer Science*, pp. 406–424, 2009.

[21] A. L. Selvakumar and C. S. Ganadhas, "The evaluation report of sha-256 crypt analysis hash function," in *Proceedings of the International Conference on Communication Software and Networks (ICCSN '09)*, pp. 588–592, June 2009.

[22] B. Baldwin, A. Byrne, M. Hamilton et al., "FPGA implementations of SHA-3 candidates: cubehash, grostl, lane, shabal and spectral hash," in *Proceedings of the 12th Euromicro Conference on Digital System Design: Architectures, Methods and Tools, (DSD '09)*, pp. 783–790, Patras, Greece, August 2009.

[23] N. Sklavos and P. Kitsos, "Blake hash function family on fpga: from the fastest to the smallest," in *Proceedings of the 2010 IEEE Computer Society Annual Symposium on VLSI (ISVLSI '10)*, pp. 139–142, September 2010.