PAMA: A Proactive Approach to Mitigate False Data Injection Attacks in Smart Grids

Beibei Li[†,‡], Rongxing Lu[‡], Gaoxi Xiao[†], Zhou Su[$], and Ali Ghorbani[‡]
[†]School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798
[‡]Canadian Institute for Cybersecurity, Faculty of Computer Science, University of New Brunswick, Fredericton, NB, Canada
[$]School of Mechatronic Engineering and Automation, Shanghai University, Shanghai, China 200072
Email: bli012@e.ntu.edu.sg; rlu1@unb.ca; egxxiao@ntu.edu.sg; zhousu@ieee.org; ghorbani@unb.ca

***Abstract*—The pervasiveness of information and communications technologies as well as intelligent electronic devices leads to an expanded attack surface in smart grids, making it increasingly challenging to withstand the high-profile false data injection (FDI) attacks. In this paper, we propose a Proactive Approach to Mitigate FDI Attacks (PAMA) in smart grids. With PAMA scheme, the critical information - power grid connections and configurations as well as the original measurement data - used for constructing FDI attacks is well protected from leakage or theft, so that FDI attacks are effectively mitigated. Specifically, we transform the state estimation and FDI detection application into a distributed one equipped with converted information from the critical information provided by the control center. In addition, the original measurement data is also protected by using a secure hybrid Paillier cryptosystem. Our PAMA scheme is proved to be secure and effective in mitigating FDI attacks on smart grids. The computational complexity and the communication overhead are evaluated on the standard IEEE 14-bus test system.**

***Keywords*—Smart grids, false data injection (FDI) attack, Paillier cryptosystem, state estimation.**

I. Introduction

As an expected revolutionary alternative to the existing power grids, smart grids can achieve a more resilient, efficient, and automated power delivery and control to meet the growing demands of the 21st century [1]. However, despite these promising benefits, smart grids are vulnerable to increasingly complicated threat vectors, with the proliferation of information and communications technologies (ITCs) as well as intelligent electronic devices (IEDs) across the power infrastructure [2]. In April 2016, Symantec's Internet Security Threat Report showed that the energy sector was one of the top targeted industries in spear-phishing attacks [3]. Most recently, the U.S. government issued a public warning that sophisticated threat-actors are targeting industrial firms [4]. It might be a certainty that we will see an uptick in malignant cyber attacks against national critical infrastructures in the next years.

Regardless of the security or power community, one of the most popular research focuses in smart grids security lies in the high-profile false data injection (FDI) attacks. To construct FDI attacks, the attackers need to know the critical information of power grid connections and configurations as well as the access to the original measurement data. The full integration of ITCs and IEDs leaves an increasing number of vulnerabilities and interfaces outside of the power grids, providing attackers with more opportunities to collect critical information and measurement data. Further, it was reported that an underground economy has been created on the Dark Web to buy, sell, and repurpose new exploits from National Security Agency (NSA) and Central Intelligence Agency (CIA) leaks. These exploits will facilitate the success of FDI attacks, and inversely, make it increasingly challenging for defenders to build effective defenses against such cyber attacks [5].

To meet this gap, in this work, we propose a proactive approach to mitigate FDI attacks (PAMA) in smart grids. Unlike conventional approaches with focuses on post-forensics and passive detection of FDI attacks after they take place and cause damages [1], our focus is on proactive mitigation and prevention of FDI attacks before they construct and launch. Specifically, we design a distributed computing model integrated with Paillier cryptosystem to hide the information of power grid connections and configurations as well as the original measurement data, and further to mitigate the construction of FDI attacks. The main contributions of this work are two-fold:

- First, we propose a novel proactive approach to mitigate and prevent FDI attacks in smart grids, which outweighs most of existing post-forensics and passive detection schemes. We hope this work can serve as a landmark for further studies to contain FDI attacks on smart grids.
- Second, we design a distributed computing model for future smart grids to execute state estimation and FDI detection applications, which can significantly reduce the control center's computational cost. In addition to state estimation and FDI detection, this computing model can also be applied for other purposes, especially for (near) real-time applications.

The remainder of this paper is organized as follows. In Section II, we introduce our system model, adversarial model and design goal. Then, we describe some preliminaries in Section III. In Section IV, we present our proposed scheme, followed by security analysis and performance evaluation in Section V and Section VI, respectively. Finally, we draw our conclusion in Section VII.

II. Models and Design Goals

In this section, we formalize our system model, adversial model, and identify our design goals.

A. System Model

In our system model, we consider a general wide area measurement and control system (WAMCS) in smart grids [6], which includes four types of entities including a control center, a FDI detection module associated with the control center, a set of phasor data concentrators (PDCs) $\mathcal{V} = \{V_1, V_2, \cdots, V_\delta\}$, and a set of phasor measurement units (PMUs) $\mathcal{U} = \{U_1, U_2, \cdots, U_l\}$, as shown in Fig. 1. Assume that the smart grid infrastructure is physically divided into δ regions $\mathcal{R} = \{R_1, R_2, \cdots, R_\delta\}$, each $R_k \in \mathcal{R}$ has one PDC V_k and l_k PMUs. Then, the total numbers of PDCs and PMUs in our system model are δ and $l = \sum_{k=1}^{\delta} l_k$, respectively.

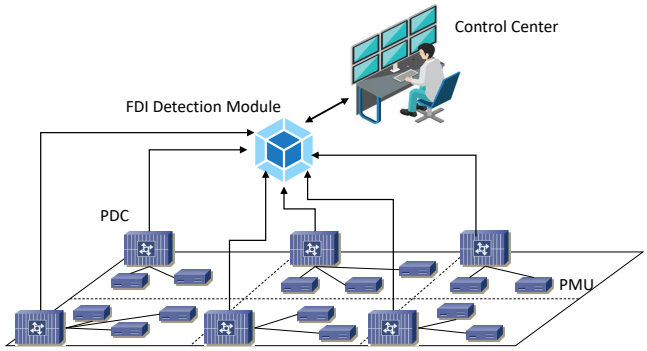

Fig. 1. System model under consideration

- PMUs $\mathcal{U} = \{U_1, U_2, \cdots, U_l\}$: Each PMU $U_i \in \mathcal{U}$ periodically collects synchronised measurement data of power grid operating status (e.g., power flows, currents, voltages) in a real-time mode with a usual frequency of 50Hz. These measurement data are then reported to a PDC in the region where the PMU U_i is deployed. Note that $V_k(U_i)$ denotes the PDC V_k that is located in the same region R_k with PMU U_i.
- PDCs $\mathcal{V} = \{V_1, V_2, \cdots, V_\delta\}$: Each PDC $V_k \in \mathcal{V}$ preprocesses the reported measurement data from all l_k PMUs in region R_k, then aggregates and relays these preprocessed data to the FDI detection module.
- FDI detection module: This module, which is integrated with a state estimator and a false data detector, is responsible for secure state estimation and FDI detection. An output of FDI detection result will be informed to the control center.
- Control center: The control center is a decision-maker, which determines what operations are to be carried out fed with outputs of various application modules, such as FDI detection, contingency analysis, etc.

Concretely, in our system model, the FDI detection module makes use of the conventional state estimation to enable FDI detection. In a direct current (DC) power flow model, given the reported measurement data vector $\mathbf{z} \in \mathbb{R}^{ld\times 1}$, the relationship between $\mathbf{z}$ and the system state vector $\mathbf{x} \in \mathbb{R}^{l\times 1}$ can be described as [7]

$$\mathbf{z} = \mathbf{H}\mathbf{x} + \boldsymbol{\eta}, \tag{1}$$

where $\mathbf{H} \in \mathbb{R}^{ld\times l}$ is the measurement Jacobian matrix containing the critical information of the power grid connections and configurations, and $\boldsymbol{\eta} \in \mathbb{R}^{ld\times 1}$ is the measurement noise vector with zero mean and covariance $\mathbf{W} = \text{diag}(\sigma_1^2, \sigma_2^2, \cdots, \sigma_{ld}^2) \in \mathbb{R}^{ld\times ld}$. Note that d is the dimension of measurement data that a PMU collects, and σ_i^2 is the variance for each measurement. With Eq. (1), the estimated system status vector is given by

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} (\mathbf{z} - \mathbf{H}\mathbf{x})^\intercal \mathbf{W}^{-1} (\mathbf{z} - \mathbf{H}\mathbf{x}). \tag{2}$$

The solution can be obtained through a non-iterative procedure by solving Eq. (2), which is given by

$$\hat{\mathbf{x}} = (\mathbf{H}^\intercal \mathbf{W}^{-1} \mathbf{H})^{-1} \mathbf{H}^\intercal \mathbf{W}^{-1} \mathbf{z}. \tag{3}$$

The estimated measurement data $\hat{\mathbf{z}}$ is then given by $\hat{\mathbf{z}} = \mathbf{H}\hat{\mathbf{x}}$. Given the estimated measurement data $\hat{\mathbf{z}}$, the false data detector at the FDI detection module calculates the normalized measurement residual vector $\boldsymbol{\gamma} \in \mathbb{R}^{ld\times 1}$ by

$$\boldsymbol{\gamma} = \sqrt{\mathbf{W}^{-1}}(\mathbf{z} - \hat{\mathbf{z}}) = \sqrt{\mathbf{W}^{-1}}(\mathbf{z} - \mathbf{H}\hat{\mathbf{x}}). \tag{4}$$

Then, the Frobenius norm of $\boldsymbol{\gamma}$ is compared with a predefined threshold τ to detect FDI attacks, i.e.,

$$\begin{cases} \text{Existence of FDI attack,} & \text{if } \|\boldsymbol{\gamma}\| > \tau, \\ \text{Non-existence of FDI attack,} & \text{otherwise.} \end{cases} \tag{5}$$

B. Adversarial Model

In the adversarial model, we consider FDI attacks in smart grids where attackers attempt to inject falsified measurement data to mislead state estimation and blind FDI detection. With the obtained knowledge of $\mathbf{H}$ matrix, the attackers can construct an attack vector $\mathbf{a} \in \mathbb{R}^{ld\times 1}$ by $\mathbf{a} = \mathbf{H}\mathbf{c}$, and fabricate a malicious measurement data vector by $\mathbf{z_a} = \mathbf{z} + \mathbf{a}$. Note that $\mathbf{c} \in \mathbb{R}^{l\times 1}$ is a theoretically arbitrary vector, indicating the offsets of power system states that FDI attackers desire to inject. In this case, the estimated system state vector given $\mathbf{z_a}$, with reference to Eq. (3), is now represented by

$$\hat{\mathbf{x}}_\mathbf{a} = (\mathbf{H}^\intercal \mathbf{W}^{-1} \mathbf{H})^{-1} \mathbf{H}^\intercal \mathbf{W}^{-1} \mathbf{z_a} = \hat{\mathbf{x}} + \mathbf{c}. \tag{6}$$

Then, the Frobenius norm of $\boldsymbol{\gamma_a}$ is given by [7]

$$\begin{aligned} \|\boldsymbol{\gamma_a}\| &= \|\sqrt{\mathbf{W}^{-1}}(\mathbf{z_a} - \mathbf{H}\hat{\mathbf{x}}_\mathbf{a})\| \\ &= \|\sqrt{\mathbf{W}^{-1}}[|\mathbf{z} + \mathbf{a} - \mathbf{H}(\hat{\mathbf{x}} + \mathbf{c})]\| \\ &= \|\sqrt{\mathbf{W}^{-1}}[\mathbf{z} - \mathbf{H}\hat{\mathbf{x}} + (\mathbf{a} - \mathbf{H}\mathbf{c})]\| \\ &= \|\sqrt{\mathbf{W}^{-1}}(\mathbf{z} - \mathbf{H}\hat{\mathbf{x}})\| \leq \tau. \end{aligned} \tag{7}$$

By checking with Eq. (5), it is clear that no FDI attack is detected in this case. As we can see, the adversaries can launch FDI attacks if they have the knowledge of $\mathbf{H}$ matrix and the access of measurement data $\mathbf{z}$. It is usually assumed that the attackers may have the capabilities of compromising a small number PMUs, the capabilities of compromising an even smaller number of PDCs, possible access to some measurement data via communication links, as well as possible knowledge of $\mathbf{H}$ matrix stored in the control center [7].

C. Design Goals

In this work, our design goal is to propose a proactive approach to mitigate FDI attacks in smart grids. Concretely,

- We aim to hide $\mathbf{H}$ matrix so that even if the attackers have some access to the control center or FDI detection module, they still cannot obtain $\mathbf{H}$ matrix. Meanwhile, we need to guarantee the execution of state estimation and FDI detection application after hiding $\mathbf{H}$ matrix.
- We also plan to hide the plaintext of the original measurement data $\mathbf{z}$ at the very front data transmission side - the PMUs - so that attackers cannot easily access and falsify $\mathbf{z}$ through links of PMU-to-PDC communication and PDC-to-FDI detection module communication.
- Unlike conventional centralized approaches for state estimation and FDI detection, our proposal - motivated also by distributed computing - attempts to reduce the computational cost of the control center by degrading the computation tasks to PDCs and the FDI detection module.

III. Preliminaries

In this section, we briefly review the original Paillier public key encryption (PKE) [8] and introduce a hybrid Paillier PKE, which will serve as building blocks of our proposed scheme.

A. Original Paillier PKE

Paillier PKE is a popular homomorphic encryption technique and comprised of three algorithms, namely the key generation *KeyGen(κ)*, encryption *Enc($\mathcal{PK}, m$)*, and decryption *Dec($\mathcal{SK}, c$)*.

- *KeyGen(κ)*: Given the security parameter $\kappa \in \mathbb{Z}^+$, choose two large prime numbers p and q such that the bit length $|p| = |q| = \kappa$. Then, $n = pq$ and $\lambda = lcm(p-1, q-1)$ are computed. Pick a generator $g \in \mathbb{Z}^*_{n^2}$, and compute $\mu = (L(g^\lambda \bmod n^2))^{-1} \bmod n$, where function L is defined as $L(x) = (x-1)/n$. Finally, *KeyGen(κ)* outputs $\mathcal{PK} = (n, g)$ as the public key, and $\mathcal{SK} = (\lambda, \mu)$ as the corresponding private key.
- *Enc($\mathcal{PK}, m$)*: Given a message $m \in \mathbb{Z}_n$, select a random number $r \in \mathbb{Z}^*_n$, and compute the ciphertext $c = E(m, r) = g^m \cdot r^n \bmod n^2$.
- *Dec($\mathcal{SK}, c$)*: Given the ciphertext $c \in \mathbb{Z}^*_{n^2}$, the plaintext m can be recovered by $m = L(c^\lambda \bmod n^2) \cdot \mu \bmod n$.

Homomorphic Properties. The Paillier PKE enjoys the following two homomorphic properties.

- *Homomorphic Addition:* Given two ciphertexts $E(m_1, r_1) = g^{m_1} \cdot {r_1}^n \bmod n^2$ and $E(m_2, r_2) = g^{m_2} \cdot {r_2}^n \bmod n^2$, we have $E(m_1, r_1) \cdot E(m_2, r_2) = E(m_1 + m_2, r_1 r_2)$.
- *Homomorphic Multiplication:* Given a ciphertext $E(m_1, r_1)$ and a plaintext $m_2 \in \mathbb{Z}_n$, we have $E(m_1, r_1)^{m_2} = E(m_1 \cdot m_2, r_1^{m_2})$.

B. Hybrid Paillier PKE

Hybrid Paillier PKE is a homomorphic encryption technique, which is comprised of four algorithms, namely the key generation, encryption, decryption-I, and decryption-II.

- *Key generation:* The key generation is similar as that in the original Paillier PKE, which outputs $\mathcal{PK} = (n, g)$ as the public key, and $\mathcal{SK} = (\lambda, \mu)$ as the corresponding private key. Different from the original Paillier PKE, we set the generator $g = n + 1$. In addition, a cryptographic hash function $H : \{0,1\}^* \to \mathbb{Z}^*_n$ is chosen, and a random number s is selected as the symmetric key shared between the encryptor and the decryptor for the hybrid Paillier PKE.
- *Encryption:* In order to encrypt a message $m \in \mathbb{Z}_n$, the encrypter chooses a random number $r \in \mathbb{Z}_n$, and computes $c = g^m \cdot H(r)^{sn} \bmod n^2$. Then, the encrypter sets (c, r) as the ciphertext.
- *Decryption-I:* If the decryptor has the private key $\mathcal{SK} = (\lambda, \mu)$, he can recover the message m by computing $m = L(c^\lambda \bmod n^2) \cdot \mu \bmod n$.
- *Decryption-II:* If the decryptor has the symmetric key s, he can also recover the message m by the following steps:
  - Because $g = n + 1$, we have $c = g^m \cdot H(r)^{sn} \bmod n^2 = (n+1)^m \cdot H(r)^{sn} \bmod n^2$. Then, the decryptor computes $H(r)^{sn}$ and obtains $c' = \frac{c}{H(r)^{sn}} \bmod n^2 = (n+1)^m \bmod n^2$.
  - Because we can easily see that $c' = (n+1)^m \bmod n^2 = nm + 1$, the decryptor can recover m by computing $L(c') = \frac{c'-1}{n} = \frac{nm+1-1}{n} = m$.

Obviously, the hybrid Paillier PKE also enjoys the two homomorphic properties.

IV. The Proposed PAMA Scheme

In this section, we present the proposed PAMA scheme to mitigate the FDI attacks in smart grids. Before delving into the details, we first show the rationale of PAMA.

A. The Rationale of PAMA

In order to achieve our goal of hiding $\mathbf{H}$ matrix from attackers, we rewrite Eq. (4) as

$$\begin{aligned}\boldsymbol{\gamma} &= \sqrt{\mathbf{W}^{-1}}(\mathbf{z} - \mathbf{H}\hat{\mathbf{x}}) \\ &= \sqrt{\mathbf{W}^{-1}}[\mathbf{I} - \mathbf{H}(\mathbf{H}^\intercal\mathbf{W}^{-1}\mathbf{H})^{-1}\mathbf{H}^\intercal\mathbf{W}^{-1}]\mathbf{z} \triangleq \boldsymbol{\Omega}\mathbf{z},\end{aligned} \tag{8}$$

where $\boldsymbol{\Omega} = \sqrt{\mathbf{W}^{-1}}[\mathbf{I} - \mathbf{H}(\mathbf{H}^\intercal\mathbf{W}^{-1}\mathbf{H})^{-1}\mathbf{H}^\intercal\mathbf{W}^{-1}] \in \mathbb{R}^{ld \times ld}$. Containing information of only $\mathbf{H}$ and $\mathbf{W}$ matrices, $\boldsymbol{\Omega}$ can be calculated in advance to support state estimation and FDI detection application. More importantly, $\boldsymbol{\Omega}$ also enables protection of the secrecy of $\mathbf{H}$ matrix because one cannot derive $\mathbf{H}$ directly from $\boldsymbol{\Omega}$. In this way, the control center only needs to store the $\boldsymbol{\Omega}$ matrix in its database. The $\mathbf{H}$ matrix is, therefore, well protected regardless of any illicit (but limited) access to the control center.

Further, as Eqs. (9) and (10) show, matrices $\boldsymbol{\Omega}$ and $\mathbf{z}$ can be partitioned into several blocks, where $\boldsymbol{\omega}_i \in \mathbb{R}^{ld \times d}$ and $\mathbf{z}_i \in \mathbb{R}^{d \times 1}$ for $1 \le i \le l$. Then $\boldsymbol{\gamma}$ can be rewritten as

$$\boldsymbol{\gamma} = \boldsymbol{\Omega}\mathbf{z} = \boldsymbol{\omega}_1\mathbf{z}_1 + \boldsymbol{\omega}_2\mathbf{z}_2 + \cdots + \boldsymbol{\omega}_l\mathbf{z}_l. \tag{11}$$

As we see, the calculation of $\boldsymbol{\gamma}$ can be partitioned into l parts, making it possible to achieve distributed computing of $\boldsymbol{\gamma}$. It is

$$\mathbf{\Omega} = \left[\begin{array}{cccc:cccc:c:ccc} \omega_{1,1} & \omega_{1,2} & \cdots & \omega_{1,d} & \omega_{1,d+1} & \cdots & \omega_{1,2d} & \cdots & \omega_{1,(l-1)d+1} & \cdots & \omega_{1,ld} \\ \omega_{2,1} & \omega_{2,2} & \cdots & \omega_{2,d} & \omega_{2,d+1} & \cdots & \omega_{2,2d} & \cdots & \omega_{2,(l-1)d+1} & \cdots & \omega_{2,ld} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \cdots & \vdots & \ddots & \vdots \\ \omega_{ld,1} & \omega_{ld,2} & \cdots & \omega_{ld,d} & \omega_{ld,d+1} & \cdots & \omega_{ld,2d} & \cdots & \omega_{ld,(l-1)d+1} & \cdots & \omega_{ld,ld} \end{array}\right] = (\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \cdots, \boldsymbol{\omega}_l) \quad (9)$$

$$\mathbf{z} = (\ z_{1,1} \quad z_{1,2} \quad \cdots \quad z_{1,d} \ \vdots\ z_{2,1} \quad z_{2,2} \quad \cdots \quad z_{2,d} \ \vdots\ \cdots \ \vdots\ z_{l,1} \quad z_{l,2} \quad \cdots \quad z_{l,d}\ )^{\intercal} = (\mathbf{z}_1, \mathbf{z}_2, \cdots, \mathbf{z}_l)^{\intercal} \quad (10)$$

then natural for us to consider PDCs as being the distributed computing agent to undertake these computing tasks, because PDCs acting as the local control centers in future power grids have sufficient computing capabilities as well as legitimate access to PMU-reported measurement data. Specifically, each PDC $V_k(U_i)$ can compute $\boldsymbol{\omega}_i \mathbf{z}_i$ for all $i \in \mathcal{I}_k$, where $\mathcal{I}_k$ is a set of the indexes of PMUs in region R_k and $|\mathcal{I}_k| = l_k$. Then, the results of distributed computing at all the PDCs are aggregated and reported to the FDI detection module for state estimation and FDI detection application.

B. Description of PAMA

Now, we present our proposed PAMA scheme, which consists of four phases: System Initialization, Measurement Data Report Generation by PMUs, Encrypted Measurement Data Preprocessing by PDCs, and Secure FDI Detection by FDI Detection Module.

1) System Initialization: We assume that the control center is a centralized party that can bootstrap the whole communication system. In our PAMA scheme, we use the hybrid Paillier cryptosystem to achieve proactive mitigation of FDI attacks. Specifically, in the system initialization phase, the control center generates the public key $\mathcal{PK} = (n, g)$ and private key $\mathcal{SK} = (\lambda, \mu)$, choose a cryptographic hash function $H : \{0,1\}^* \to \mathbb{Z}_n^*$, and publishes $\mathcal{PK}$ and H. Next, the control center needs to distribute the necessary key materials to each PMU $U_i \in \mathcal{U}$, each PDC $V_k \in \mathcal{V}$, and the FDI detection module by the following steps:

- ***Step-1***: Define a function $\tilde{\nu} = f(\nu) = \lfloor 1000 \cdot \nu \rfloor \bmod n$, and apply the function on $\mathbf{\Omega}$ so that each element $\omega_{i,j} \in \mathbf{\Omega}$, with $1 \le i, j \le ld$, is converted into an integer in $\mathbb{Z}_n$. Then, partition $\mathbf{\Omega}$ matrix into $\mathbf{\Omega} = \{\tilde{\boldsymbol{\omega}}_i | 1 \le i \le l\}$ as Eq. (9), and distribute them to each PDC $V_k(U_i)$ with $1 \le k \le \delta$, respectively.
- ***Step-2***: Choose $l \times d$ random numbers $s_{i,j} \in \mathbb{Z}_n^*$, with $1 \le i \le l$ and $1 \le j \le d$, to form the key set $\mathbf{s} = \{s_{i,j} \in \mathbb{Z}_n^* | 1 \le i \le l, 1 \le j \le d\}$.
- ***Step-3***: For each PMU U_i, the control center assigns the secret key $\mathbf{s}_i = \{s_{i,1}, s_{i,2}, \cdots, s_{i,d}\} \subseteq \mathbf{s}$ to U_i. Correspondingly, the control center also distributes $\tilde{\boldsymbol{\omega}}_i$ to PDC $V_k(U_i)$.
- ***Step-4***: Finally, the control center computes ld secret keys

$$sk_\zeta = \sum_{i=1}^{l} \sum_{j=1}^{d} \tilde{\omega}_{\zeta,(i-1)d+j} \cdot s_{i,j} \bmod n \quad (12)$$

for $1 \le \zeta \le ld$, and assigns these ld secret keys $(sk_1, sk_2, \cdots, sk_{ld})$ all to the FDI detection module.

2) Measurement Data Report Generation by PMUs : PMUs $\mathcal{U}$ periodically report the measurement data to the control center via PDCs. Specifically, at each epoch time t, each PMU $U_i \in \mathcal{U}$ collects d types of measurement data $\mathbf{z}_i = (z_{i,1}, z_{i,2}, \cdots, z_{i,d})^{\intercal}$ in either positive or negative decimals or zeros, and performs the following steps:

- ***Step-1***: Magnify the original measurement data $\mathbf{z}_i$ by computing $\tilde{\mathbf{z}}_i = f(\mathbf{z}_i)$ to ensure that all the vector entries are non-negative integers in $\mathbb{Z}_n$, i.e.,

$$\begin{aligned} \tilde{\mathbf{z}}_i = f(\mathbf{z}_i) &= \lfloor 1000 \times \mathbf{z}_i \rfloor \bmod n \\ &= (\tilde{z}_{i,1}, \tilde{z}_{i,2}, \cdots, \tilde{z}_{i,d})^{\intercal} \bmod n \end{aligned} \quad (13)$$

- ***Step-2***: Compute $H(t)$ and encrypt each dimension of measurement data $\tilde{z}_{i,j}$, $1 \le j \le d$, with the secret keys $\mathbf{s}_i = \{s_{i,1}, s_{i,2}, \cdots, s_{i,d}\}$ as

$$E(\tilde{z}_{i,j}) = g^{\tilde{z}_{i,j}} \cdot H(t)^{n \cdot s_{i,j}} \bmod n^2. \quad (14)$$

Then, the encrypted measurement data $E(\tilde{\mathbf{z}}_i)$ of all dimensions ($1 \le j \le d$) for PMU U_i is given by

$$\begin{aligned} E(\tilde{\mathbf{z}}_i) &= E[(\tilde{z}_{i,1}, \tilde{z}_{i,2}, \cdots, \tilde{z}_{i,d})]^{\intercal} \\ &= \begin{pmatrix} g^{\tilde{z}_{i,1}} \cdot H(t)^{n \cdot s_{i,1}} \bmod n^2 \\ g^{\tilde{z}_{i,2}} \cdot H(t)^{n \cdot s_{i,2}} \bmod n^2 \\ \vdots \\ g^{\tilde{z}_{i,d}} \cdot H(t)^{n \cdot s_{i,d}} \bmod n^2 \end{pmatrix} \end{aligned} \quad (15)$$

- ***Step-3***: Report the encrypted measurement data $E(\tilde{\mathbf{z}}_i)$ to PDC $V_k(U_i)$.

3) Encrypted Measurement Data Preprocessing by PDCs: The PDC V_k in region R_k performs the following steps:

- ***Step-1***: For each PMU U_i in region R_k, PDC V_k preprocesses the encrypted measurement data $E(\tilde{\mathbf{z}}_i)$ with $\tilde{\boldsymbol{\omega}}_i$ by computing $\tilde{Z}_{i,\zeta}$, where $1 \le \zeta \le ld$, as

$$\begin{aligned} \tilde{Z}_{i,\zeta} &= \prod_{j=1}^{d} E(\tilde{z}_{i,j})^{\tilde{\omega}_{\zeta,(i-1)d+j}} \bmod n^2 \\ &= g^{\sum_{j=1}^{d} \tilde{\omega}_{\zeta,(i-1)d+j} \cdot \tilde{z}_{i,j}} \times \\ &\quad H(t)^{n \cdot \sum_{j=1}^{d} \tilde{\omega}_{\zeta,(i-1)d+j} \cdot s_{i,j}} \bmod n^2 \end{aligned} \quad (16)$$

Then, the preprocessed data for all dimensions ($1 \le \zeta \le ld$) regarding PMU U_i are indicated by $\tilde{\mathbf{Z}}_i = (\tilde{Z}_{i,1}, \tilde{Z}_{i,2}, \cdots, \tilde{Z}_{i,ld})^{\intercal}$ at the PDC V_k.

- ***Step-2***: For all $\tilde{\mathbf{Z}}_i$ with $i \in \mathcal{I}_k$ and for each dimension $1 \le \zeta \le ld$, $\tilde{\mathbf{Z}}_i$ can be further aggregated by

$$\begin{aligned}\tilde{C}_{k,\zeta} &= \prod_{i\in\mathcal{I}_k} \tilde{Z}_{i,\zeta} \\ &= g^{\sum_{i\in\mathcal{I}_k}\sum_{j=1}^{d}\tilde{\omega}_{\zeta,(i-1)d+j}\cdot\tilde{z}_{i,j}}\times \\ &\quad H(t)^{n\cdot\sum_{i\in\mathcal{I}_k}\sum_{j=1}^{d}\tilde{\omega}_{\zeta,(i-1)d+j}\cdot s_{i,j}} \bmod n^2\end{aligned} \tag{17}$$

  Then, the aggregated data for all dimensions ($1 \le \zeta \le ld$) are given by $\tilde{\mathbf{C}}_k = (\tilde{C}_{k,1}, \tilde{C}_{k,2}, \cdots, \tilde{C}_{k,ld})^\intercal$.
- ***step-3***: Report the aggregated preprocessed measurement data $\tilde{\mathbf{C}}_k$ to the FDI detection module.

4) Secure FDI Detection by FDI Detection Module: In this phase, the following steps will be performed for FDI detection.

- ***Step-1***: Given $\tilde{\mathbf{C}}_k$ from each region R_k with $1 \le k \le \delta$, compute $\tilde{\Gamma}_\zeta$, which is the ζ-th element of the encrypted measurement residual, for $1 \le \zeta \le ld$, by

$$\begin{aligned}\tilde{\Gamma}_\zeta = \prod_{k=1}^{\delta}\tilde{C}_{k,\zeta} &= g^{\sum_{i=1}^{l}\sum_{j=1}^{d}\tilde{\omega}_{\zeta,(i-1)d+j}\cdot\tilde{z}_{i,j}}\times \\ &\quad H(t)^{n\cdot\sum_{i=1}^{l}\sum_{j=1}^{d}\tilde{\omega}_{\zeta,(i-1)d+j}\cdot s_{i,j}} \bmod n^2.\end{aligned} \tag{18}$$

- ***Step-2***: Given the secret keys $(sk_1, sk_2, \cdots, sk_{ld})$ and the hash function H, compute $H(t)^{n\cdot sk_\zeta} \bmod n^2$ for all $1 \le \zeta \le ld$. Then, decrypt $\tilde{\Gamma}_\zeta$ for all $1 \le \zeta \le ld$ as

$$\begin{aligned}\tilde{\gamma}_\zeta &= L\Big(\frac{\tilde{\Gamma}_\zeta}{H(t)^{n\cdot sk_\zeta}} \bmod n^2\Big) \\ &= \sum_{i=1}^{l}\sum_{j=1}^{d}\tilde{\omega}_{\zeta,(i-1)d+j}\cdot\tilde{z}_{i,j} \bmod n.\end{aligned} \tag{19}$$

  The aggregated $\tilde{\boldsymbol{\gamma}}$, which is the plaintext normalized measurement residual vector, is therefore given by $\tilde{\boldsymbol{\gamma}} = (\tilde{\gamma}_1, \tilde{\gamma}_2, \cdots, \tilde{\gamma}_{ld})^\intercal$.
- ***Step-3***: Compute the sum of squares of $\tilde{\gamma}_\zeta$ for all $1 \le \zeta \le ld$ modulo n by

$$\rho = \sum_{\zeta=1}^{ld}\tilde{\gamma}_\zeta^2 \bmod n. \tag{20}$$

- ***Step-4***: Since $\tilde{\omega}_{\zeta,(i-1)d+j}$ and $\tilde{z}_{i,j}$ for computing $\tilde{\gamma}_\zeta$ with all $1 \le \zeta \le ld$, $1 \le i \le l$, and $1 \le j \le d$ are magnified by 1000 times, to compare the Frobenius norm of the original normalized measurement residual $\|\boldsymbol{\gamma}\| = \sqrt{\sum_{\zeta=1}^{ld}\gamma_\zeta^2}$ to the given threshold τ for FDI detection, we also magnify τ by $\tilde{\tau} = (1000\times1000)^2\times\tau^2$ times and compare it to ρ:

$$\begin{cases}\text{Existence of FDI attack,} & \text{if } \rho > \tilde{\tau}, \\ \text{Non-existence of FDI attack,} & \text{otherwise.}\end{cases} \tag{21}$$

V. Security Analysis

In this section, we analyze the security of our proposed PAMA scheme. Specifically, under our adversarial model, we will show that both $\mathbf{H}$ and $\mathbf{z}$ are protected, and PAMA can really mitigate the FDI attack in smart grids.

A. Secrecy Preservation of $\mathbf{H}$ Matrix

In our PAMA scheme, the control center calculates an $\boldsymbol{\Omega}$ matrix and distributes its l partitions to the PDCs, instead of giving $\mathbf{H}$ matrix directly to the FDI detection module for state estimation and FDI detection. One the one hand, one cannot derive $\mathbf{H}$ directly from $\boldsymbol{\Omega}$. Hence, even if attackers can obtain $\boldsymbol{\Omega}$ with certain access to the control center, they cannot recover $\mathbf{H}$. On the other hand, although each PDC $V_k(U_i)$ has l_k partitions $\{\boldsymbol{\omega}_i | i \in \mathcal{I}_k\}$ of the $\boldsymbol{\Omega}$ matrix in hand, it is reasonably assumed, as mentioned in our adversarial model, that attackers are incapable of compromising all the PDCs; therefore, they cannot get these full partitions $\{\boldsymbol{\omega}_i | 1 \le i \le l\}$ to recover $\boldsymbol{\Omega}$ through the compromised PDCs. In this way, the secrecy of $\mathbf{H}$ matrix is well preserved from malicious access by using our PAMA scheme.

B. Secrecy Preservation of Original Measurement Data $\mathbf{z}$

The secrecy of original measurement data $\mathbf{z}$ is preserved mainly by using the secure hybrid Paillier PKE. $\mathbf{z}$ is encrypted before being transmitted to a PDC for data preprocessing and further to the FDI detection module for state estimation and FDI detection. Even if the attackers can compromise the PDCs and/or the FDI detection module or intercept with the communication links in between, they still cannot recover the original measurement data $\mathbf{z}$ without secret keys $s_{i,j} \in \mathbf{s}$ or the private key $\mathcal{SK}$. In addition, due to limited capabilities as assumed in our adversarial model, the attackers are capable of compromising only a small number of PMUs; thus, only limited original measurement data may be leaked.

As we can see from the above analysis, our proposed PAMA scheme can proactively preserve the secrecy of both $\mathbf{H}$ matrix and original measurement data $\mathbf{z}$, which are significantly important to construct FDI attacks. As a result, PAMA can effectively mitigate the FDI attacks by hiding $\mathbf{H}$ and $\mathbf{z}$ from malicious access.

VI. Performance Evaluation

In this section, we evaluate the performance of our proposed PAMA scheme in terms of the computational complexity of each PMU, PDC, and the FDI detection module, respectively, as well as the communication overhead of PMU-to-PDC and PDC-to-FDI detection module communications. In our simulations, we set $\kappa = 512$, thus $|p| = |q| = 512$ bits, and $|n| = 1024$ bits. Our simulations are conducted on an Intel(R) Core(TM) i7-7700 CPU @3.60GHz Windows Platform with 16GB RAM. We simulated $10,000$ times of secure state estimation and FDI detection on the standard IEEE 14-bus test system for each case, respectively, where different settings of the number of PMUs (l) and the dimension of measurement data (d) are applied. It is assumed that the standard IEEE 14-bus test system is divided into 5 regions, each of which is located with a PDC and at least one PMU. The average results for PMU, PDC, and the FDI detection module are respectively plotted in Figs. 2, 3, and 4.

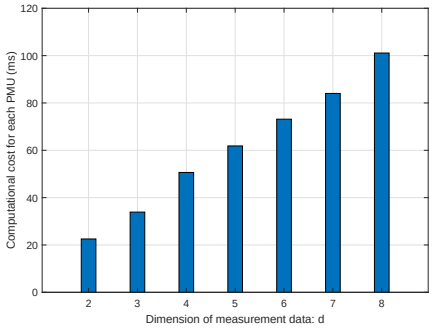

Fig. 2. The computational cost for each PMU versus d varying from 2 to 8.

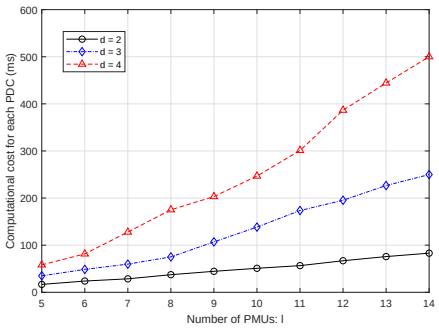

Fig. 3. The computational cost for each PDC versus l varying from 5 to 14.

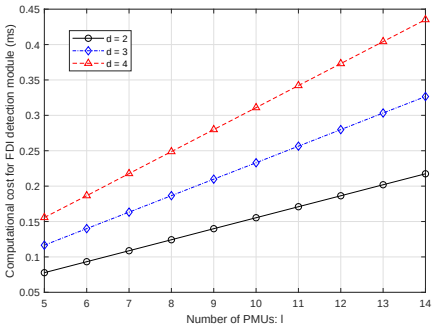

Fig. 4. The computational cost for the FDI detection module versus l varying from 5 to 14.

A. Computational Complexity

The average computational cost for each PMU to generate measurement data reports versus the dimension of measurement data d are plotted in Fig. 2. As we can see from this figure, the computational cost increases almost linearly proportionally to the dimension of measurement data. This is because that each PMU requires $2d$ exponentiation operations in $\mathbb{Z}^*_{n^2}$ to encrypt the measurement data. An increased dimension of measurement data, therefore, leads to an increased number of encryption operations, which increases the computational cost.

Fig. 3 plots the average computational cost for each PDC versus the number of PMUs l under different ds. As can be seen, the computational cost increases almost linearly proportionally to l and almost quadratically to d. The reason is that a PDC on average conducts $\mu \times l \times d^2$ exponentiation operations in $\mathbb{Z}^*_{n^2}$, where μ is the average number of PMUs in a region.

In Fig. 4, the computational cost for the FDI detection module versus the number of PMUs l under different ds are plotted. It is obvious that the computational cost for the FDI detection module, mainly relying on the decryption time, increases almost linearly proportionally to both l and d, respectively. This is because that the decryption time is linearly proportionally to the size of measurement residual vector $l \times d$.

B. Communication Overhead

The communications of the proposed PAMA scheme can be divided into two main parts: PMU-to-PDC and PDC-to-FDI detection module communications. First, we consider the PMU-to-PDC communication, where PMUs report their measurement data to the local PDC. The report size is d encrypted data in $\mathbb{Z}^*_{n^2}$ for each PMU. Thus, the communication overhead from all l PMUs to PDCs is $l \times d \times 2048$ bits. When it comes to PMU-to-PDC communication where PDCs report preprocessed measurement data to the FDI detection module, the size of report for each PDC is $l \times d$ preprocessed data in $\mathbb{Z}^*_{n^2}$. In this way, the communication overhead from all δ PDCs to the FDI detection module is $\delta \times l \times d \times 2048$ bits.

VII. Conclusions

In this paper, we proposed a novel proactive approach called PAMA to mitigate FDI attacks in smart grids. Specifically, we designed a distributed computing model to decentralize the state estimation and FDI detection application, and used a secure hybrid Paillier PKE to enable distributed computing on ciphertexts. Most importantly, our PAMA scheme well preserves the secrecy of $\mathbf{H}$ matrix as well as the original measurement data $\mathbf{z}$, both of which are critical for constructing FDI attacks. Security analysis shows PAMA can really mitigate the FDI attacks, and extensive evaluation results also validate the efficiency of PAMA.

Acknowledgements

This research was partially-supported by the Ministry of Education, Singapore under contract MOE2016-T2-1-119, the Future Resilient System project (FRS) at the Singapore-ETH Centre (SEC) funded by the National Research Foundation of Singapore (NRF) under its Campus for Research Excellence and Technological Enterprise (CREATE) program, the Natural Sciences and Engineering Research (NSERC) Discovery Grants (no. Rgpin 04009), NBIF Start-Up Grant (Rif 2017-012), and the NSFC Grant (no. 61571286).

References

[1] B. Li, R. Lu, W. Wang, and K.-K. R. Choo, "DDOA: A dirichlet-based detection scheme for opportunistic attacks in smart grid cyber-physical system," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 11, pp. 2415–2425, Nov. 2016.

[2] R. Lu, X. Liang, X. Li, X. Lin, and X. Shen, "EPPA: An efficient and privacy-preserving aggregation scheme for secure smart grid communications," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 9, pp. 1621–1631, Sep. 2012.

[3] E. Kovacs, "Internet security threat report," *Symantec*, Apr. 2016, Last accessed: 04/22/2018. [Online]. Available: https://www.symantec.com/content/dam/symantec/docs/reports/istr-21-2016-en.pdf

[4] J. Finkle, "U.S. warns public about attacks on energy, industrial firms," *Reuters*, Last accessed: 04/22/2018. [Online]. Available: https://reut.rs/2xVmFrj

[5] F. Justin, "Cyber trends defenders can expect to see in 2018," *SecurityWeek*, Last accessed: 04/22/2018. [Online]. Available: https://www.securityweek.com/cyber-trends-defenders-can-expect-see-2018

[6] W. Li, *Risk assessment of power systems: models, methods, and applications*. John Wiley & Sons, 2014.

[7] Y. Liu, P. Ning, and M. K. Reiter, "False data injection attacks against state estimation in electric power grids," *ACM T. Inform. Syst. Se.*, vol. 14, no. 1, p. 13, May 2011.

[8] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Proc. Theory and Applications of Cryptographic Techniques*. Springer, Apr. 1999, pp. 223–238.