iDDAF: An Intelligent Deceptive Data Acquisition Framework for Secure Cyber-physical Systems

Md Hasan Shahriar[1], Mohammad Ashiqur Rahman[1], Nur Imtiazul Haque[1], Badrul Chowdhury[2], and Steven G Whisenant[3]

[1] Florida International University, USA,
[2] The University of North Carolina at Charlotte, USA
[3] Duke Energy, USA
[1]{mshah068,marahman,nhaqu004}@fiu.edu,[2]b.chowdhury@uncc.edu,
[3]steven.whisenant@duke-energy.com

Abstract. Internet of Things (IoT) and Cyber-Physical Systems (CPSs) are creating hybrid platforms that are becoming ubiquitous in all modern infrastructure. As complex and heterogeneous systems are getting integrated, a malicious user can have tremendous opportunities to infiltrate networks, steal sensitive information, inject cleverly crafted false data into measurements, or overwhelm networks with fake packets. Such malicious activities can prevent legitimate requests or even mislead the control center to make erroneous decisions. Agility-based defense mechanisms are robust in deceiving adversaries by randomizing the sensor data at different communication hierarchy levels. While misleading the attackers, the control center must retrieve the actual data to operate the system correctly. Existing mechanisms consider sharing the exact remapping pattern with the control center. Such direct sharing raises the concern of further attacks on them and communication overheads. Hence, we propose iDDAF, an intelligent deception defense-based data acquisition framework that leverages system-agnostic prediction and remapping model at the controller level to ensure a comprehensive security solution (CIA triad) for any hierarchical CPSs network. In this framework, the data reporting/relaying nodes randomize the associated sensor addresses/IDs and add decoy data, while the prediction mechanism at the control center reassigns the original IDs to the measurements and imputes the missing data if necessary. Hence, any reconnaissance attempt fails, artfully altered measurements turn into random data injections, making it easy to remove them as outliers. Experimental results on the standard IEEE 14 bus system show that iDDAF can detect and completely mitigate different types of cyberattacks.

Keywords: Deception defense · Cyber-physical systems · Cyberattacks.

1 Introduction

Cyber-physical systems (CPSs) integrate sensing, communication, processing, and control from cyberspace and the physical world [13]. CPSs are everywhere - in transportation networks, smart grids, medicine, water management, and so on.

A new dimension for CPSs has been opened up by the Internet of things (IoT), enabling real-time monitoring, data exchange, and optimum control. As a key element of the control system's control process, the state estimation (SE) plays a critical role in ensuring safety and proper control decisions [4,33]. In the era of heterogeneous sensor interaction, CPSs have become essential components of critical infrastructures while also creating a large attack surface for adversaries. Modern cyberattacks are so sophisticated that legacy defense techniques cannot stand up against them. By leveraging the targeted system's knowledge and state, adversaries can launch influential attacks, such as false data injection (FDI) attacks, covert attacks, zero dynamics attacks, replay attacks, and denial of service (DoS) attacks [14,32,40].

The CPSs' supervisory control and data acquisition (SCADA) devices typically lack the processing power to perform strong encryption on sensor data streams [5]. Data acquisition processes at critical infrastructures (such as power systems) are also constrained by maximum time delays. According to the IEC 61850 standard, GOOSE (generic object-oriented substation event) messages within power grid substations must be delivered with no more than a 4 ms delay [2]. Due to this limitation, high-end security may not be applied despite sufficient computational power at the substations networks.

Several studies have shown that sophisticated attackers can generate attack data, evading the existing defense mechanisms utilizing the information about the topology and states of the targeted systems [16, 24, 29, 31]. A computer worm containing 500 kilobits of code, called Stuxnet, destroyed over a thousand Iranian uranium enrichment centrifuges in 2010 [19]. US smart grids could be breached by Stuxnet-like attacks causing an estimated $1 trillion loss to the government [9]. In Ukraine, a massive power outage has been caused by Black-Energy Malware, which targets the SCADA system with a DoS attack [17].

"All warfare is based on deception. Hence, when we are able to attack, we must seem unable; when using our forces, we must appear inactive; when we are near, we must make the enemy believe we are far away; when far away, we must make him believe we are near." -— Sun tzu, the Art of War [37].

As the author said, in cyber war, deception also plays a vital role. To secure the data acquisition process in CPSs, we propose an **i**ntelligent **D**eception **D**efense-based **D**ata **A**cquisition **F**ramework (iDDAF) for any hierarchical SCADA network. The proposed framework secures the data acquisition steps by shuffling the reported sensors' addresses/IDs along with adding decoy data at each node in the hierarchy. The randomization patterns are changed periodically to obfuscate any reconnaissance attempt. Moreover, due to the deception, any attempt of coordinated stealthy cyberattack turns into a random data injection, which is easily detected and eliminated by the existing bad data detection techniques. As smart grid is a perfect example of modern CPSs, we consider a smart grid hierarchical communication network as our testbed.

As the sensor data are randomized at the network switches/nodes, the energy management system (EMS) needs to remap them to the original pattern. The existing works consider that the nodes share the randomization information with the EMS through a seed-based or a secure communication medium. In the seed-based approach, both the nodes and the EMS must be perfectly synchronized to share the remapping pattern. Failure of synchronization can lead to a different remapping pattern and direct the system to a hazardous scenario. On the other hand, sending remapping information through another dedicated communication channel further raises the question of security. Hence, we propose a novel secure and effective data-driven approach to overcome the existing concerns. In summary, the contributions of this paper are as follows:

- We implement deception as a defense in the networked control systems to defend against different cyberattacks. We design iDDAF, utilizing a prediction and a remapping model, for any CPSs hierarchical network that can misleads the stealthiest cyberattacks.
- We use a heuristic randomization algorithm to generate the deceptive IDs for the randomized sensors. An algorithm is designed to create a virtual sub-system at the nodes considering the underneath physical components to generate the decoy data supporting the ID randomization.
- We design and deploy a highly accurate model agnostic regression algorithm for estimating the state data based on previous states information. The prediction process enables the remapping algorithm to recover the original sensor measurements from the random data sequence. The remapping algorithm is defined as an optimization problem, which takes the reported (shuffled) and predicted data as inputs and finds the best possible set of recovered data, declining the malicious ones and imputing the missing measurements.
- We implement and evaluate the framework on a standard IEEE 14 bus system. The codes and data are publicly available at [1].

The rest of the paper is organized as follows: The models and objectives are explain in Section 2. We add sufficient background information in Section 3. Our proposed iDDAF is introduced in Section 4. In Section 5, we discuss the technical details of the framework. We also provide an example case study to demonstrate the data remapping process. The evaluation setup and result analysis are formulated in Section 6. The related works are discussed in Section 7. At last, we conclude the paper in Section 8.

2 Models and Design Objectives

In this section, we describe the system and threat models, along with the design objectives of iDDAF.

2.1 System Model

The future CPS network should have a hierarchical structure to minimize communication overhead and to ensure the system's stability, reliability, and efficiency [39]. Smart grids with hierarchy-based communication networks are becoming a more preferred choice as distributed generation, renewable energy, and electricity demand increase [20]. Fig. 1 illustrates a hierarchical communication network in a smart grid – a model with two layers of substation switches.

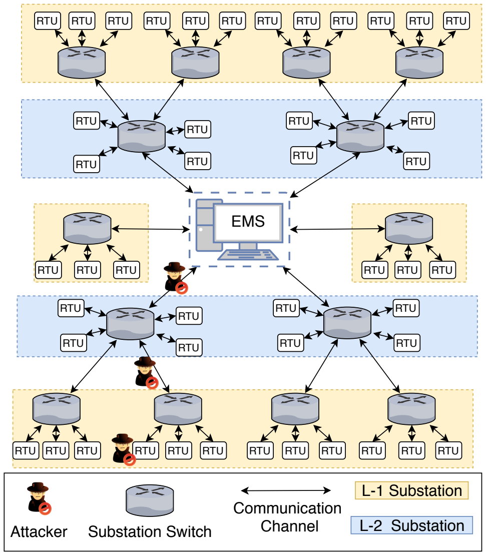


Fig. 1: Hierarchical communication network in a smart grid.

A sensor in a substation may be a remote terminal unit (RTU), an intelligent electronic device (IED), a phasor measurement unit (PMU), etc. A sensor is located within a substation and reports measurement data to its switch. Hence, the sensor are referred to as level-0 (L-0) elements of the network. A level-1 substation (L-1) receives measurement data from its own sensors only, whereas a level-2 substation (L-2) receives measurements from both its sensors and the L-1 substations underneath it. When a network has two levels, the L-2 switches report data directly to the EMS. In order to estimate the system states and make appropriate decisions after collecting remote sensor data, the EMS executes the SE-BDD (section 3.1) algorithm. The hierarchical structure can be considered like a tree, where the sensors are the leaf nodes, substation switches act as the internal nodes, and EMS is at the root. Additionally, we determine the communication channel based on the level of the incoming switch when sending data to the EMS. Therefore, a channel between L-1 and L-2 switches is defined as an L-1 channel, and a channel between the L-2 switch and EMS is an L-2 channel.

2.2 Attack Model

This subsection defines the attack model based on the attacker's abilities and goals. We consider the attack tree illustrated in Fig. 2 in this study. An adversary can exploit multiple vulnerable points of the data acquisition process. The attacker's accessibility/position into the network plays a very crucial role in the attack's success. In general, we classify an adversary's position into two groups. Firstly, the attacker can compromise the individual targeted sensors [6, 34]. In this case, the attacker is sophisticated enough to be distributed to the edge nodes of the network. Sensors are usually located within a substation, which are highly secured to be physically accessed. Since this type of attack comes with a great cost, it is less common

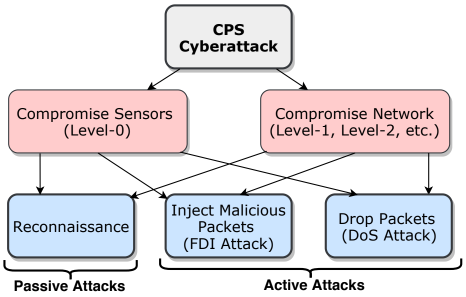

CPS Cyberattack
Compromise Sensors (Level-0)
Compromise Network (Level-1, Level-2, etc.)
Reconnaissance
Inject Malicious Packets (FDI Attack)
Drop Packets (DoS Attack)
Passive Attacks
Active Attacks

Fig. 2: Considered attack tree in CPSs.

in reality. As the sensors are the leaf nodes (L-0), we define such sensory attacks as an L-0 attack.

Secondly, the attacker can compromise network devices, like routers, switches, channels, and so on. As these elements are spread throughout the SCADA network, physical security can be compromised at some points. Moreover, high-end encryption may not be implemented in all the network switches due to low computational capabilities. Thus, compromising the communication network is more prevalent in the CPSs cyberattacks. However, network switches or routers are mostly in secured locations as they also belong to the substations/local control center. Thus, communication channels remain the most vulnerable points. When the attacker compromises L-1 communication channels, we consider them L-1 attacks; when they compromise L-2 channels, we consider them L-2 attacks.

On the other hand, depending on the attacker's intent, we again categorize the cyberattacks into two classes— active attacks, and passive attacks [35]. Passive attacks are the process of reconnaissance of the system states, where the attacker gets into the network, sniffs, and analyzes the packets without obstructing the normal operation of the system. The goal of such an attack is to study the parameters of the physical system and determine the optimal attack tactics without creating any attention of the defender. A passive attack is dangerous for the confidentiality of the system.

Active attacks are the injections of malicious data into the sensor measurements that help to achieve the attacker's goal. Active attacks exploit the integrity as well as the availability of sensor data. The effectiveness/stealthiness of the active attacks depends on the success of the passive attacks. We consider two influential cyberattacks, e.g., FDI attacks, and DoS attacks as the active attacks in the paper.

In the first case, the attacker injects malicious data into the network packets to mislead the system in a hazardous direction. In DoS attacks, the attacker drops the targeted packets, leaving the sensors from a critical part unavailable. As a result, the total system becomes unobservable and may collapse due to the delayed response. In this work, we evaluate the performance of iDDAF considering all the combination of cyberattacks as discussed.

2.3 Design Objectives

The key goals of our proposed framework is to provide a secure and robust data acquisition mechanism. Our design objectives are as follows:

- **Confidentiality**: To preserve the systems' privacy from the adversary, we hide the true information of the system and show the artfully crafted sensors data. Such move misleads the attacker and prevents reconnaissance attacks in the system states.
- **Integrity:** To maintain the unperturbed system operation, any malicious data injections need to be removed from the system control loop. Thus, we aim to mitigate the stealthy FDI injection attacks by identifying the compromised sensors and eliminate them from the SE procedure.

- **Availability:** To optimize the dynamic behaviour of the system in runtime, the system must be observable to EMS. However, in the case of DoS attacks, some parts of the system might go offline, making the whole system unobservable. We need to predict missing data and keep the system running around the optimal operating point
- To remap the randomized sensors data in agility-based defense, existing solutions consider explicit information sharing between the nodes of the network. However, our goal is to design an intelligent deceptive data acquisition framework, where the recovery process is totally independent and avoids such information sharing among the nodes.

3 Preliminaries

In this section, we discuss the terminologies that are used throughout the paper to facilitate readers' comprehension.

3.1 State Estimation and Bad Data Detector

SE is the process of determining the network's state based on redundant telemetry measurements. Let us assume that a CPS has n number of states variables, $\mathbf{x} = (x_1, x_2, \cdots, x_n)^T$, and m sensor measurements, $\mathbf{z} = (z_1, z_2, \cdots, z_m)^T$, with sufficient redundancy ($m >> n$). The states vector $\mathbf{x}$ and the measurement set $\mathbf{z}$ are related as $\mathbf{z} = \mathbf{h}(\mathbf{x}) + \mathbf{e}$, where $\mathbf{h}(\mathbf{x})$ is the function relating the measurement set to the state data, and $\mathbf{e}$ is the set of noise/errors that follows a normal distribution [4]. The state variables in a power system, for instance, are the voltages on the buses (magnitude and phase). The sensor usually measures power flow data, bus power injections, and phasor measurement units (PMUs) data, etc.

DC power flow is a widely used technique in power industries for efficient and accurate real-time analysis due to its simplicity, robustness, and high computing speed [38]. Voltage magnitude is considered a unity in the DC system approximation. So the only state variables that are considered are bus phase angles. Besides, $\mathbf{h}(\mathbf{x})$ is the linear transformation using the Jacobian matrix, $\mathbf{H}$. For the linear measurement functions, $\mathbf{h}(\mathbf{x})$ the most probable system states vector $\hat{\mathbf{x}}$ can be estimated directly by solving:

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

$\mathbf{W}$ denotes diagonal weighting matrix. Thus, the best estimated measurement vector $\mathbf{z_{est}}$ can be calculated by $\mathbf{z_{est}} = \mathbf{h}(\hat{\mathbf{x}})$ and the residuals set as $\mathbf{r} = ||\mathbf{z} - \mathbf{h}(\hat{\mathbf{x}})||$. The Bad Data Detection (BDD) procedure is often coupled with SE to identify outliers. A measurement z_i is considered as a bad data and eliminated from the SE procedure if $||r_i|| > \tau$, where τ is error threshold. The estimation and removal process is repeated until no bad data is found in the considered measurement vector. At the end, an $SE{-}BDD$. procedure returns the estimated states $\hat{\mathbf{x}}$, estimated measurements $\mathbf{z_{est}}$, residual vector $\mathbf{r}$, and the list of outliers.

3.2 Stealthy False Data Injection Attack

By injecting malicious data into the sensor reading, an attacker can alter the state estimation process. An attack vector consists of malicious data that is injected into a set of sensor measurements. So an anomaly occurs as a result of a random attacks; the compromised sensors become outliers at BDD, and the SE process is left unaffected. However, bypassing the BDD is possible if the attack vector is intelligently calculated based on the information of the targeted system [24,26]. Let us consider one scenario where the attacker wants to change the state variables set $\hat{\mathbf{x}}$ by the malicious amount $\mathbf{x^c}$. If (s)he injects false data $\mathbf{a}$ to the measurement set $\mathbf{z}$, the condition $\mathbf{a} = \mathbf{h}(\mathbf{x^c})$ ensures that such injection will bypass the BDD. Since $\mathbf{z}+\mathbf{a} = \mathbf{h}(\hat{\mathbf{x}}+\mathbf{x^c})$, the residual $\mathrm{r} = ||(\mathbf{z}+\mathbf{a})-\mathbf{h}(\hat{\mathbf{x}}+\mathbf{x^c})||$ $= ||\mathbf{z}-\mathbf{h}(\hat{\mathbf{x}})+\mathbf{a}-\mathbf{h}(\mathbf{x^c})|| = ||\mathbf{z}-\mathbf{h}(\hat{\mathbf{x}})||$. Data injection is thus rendered invisible to the residual vector, which allows the attack to remain stealthy. The attacker needs to know the topology, configuration, and measurements of the targeted system. For the rest of this paper, FDI is used to refer to stealthy FDI attacks.

4 iDDAF

The tasks of the iDDAF are categorized into two mechanisms: i) deception mechanism, and ii) remapping mechanism. The deception mechanism is implemented in the nodes of the network. As the first part of the deception, EMS assigns all the sensors into three groups: *fixed, randomized, decoyed.* The *fixed* sensors do not participate in the deception mechanism and are regarded as the regular sensors. On the other hand, *randomized* and *decoyed* sensors are considered in the deception process. EMS updates the groups periodically and lets the corresponding nodes know the sensors group information.

Later, during the data acquisition process, the nodes directly forward the packets of *fixed* sensors but craft the packets of others. They shuffle the IDs of the *randomized* sensors and add decoy data to *decoyed* sensors. The decoy data are calculated to support the SE on the *fixed* and *randomized* measurement data so that they remain topologically aligned and thus the randomization remains hidden to the attacker. This process continues until the packets reach EMS. As the nodes may need to analyze and modify the network packets, they need to be equipped with software-defined networking (SDN) controllers [8,25]. An SDN controller has the ability to read, edit, and assemble packets in real-time. SDN is widely used in modern communication networks. In the absence of SDN switches, VMware and Nicira can also implement similar capabilities on conventional switches [7]. Once EMS receives the packets, it drops off the *decoy* data and restores the remaining data along with their original IDs before using them in the SE procedure. The remapping mechanism uses a regression-based prediction model that considers the historical time-series state data to predict the next state vector. However, as EMS receives a randomly shuffled sensor data, we define the ID remapping algorithm as a combinatorial optimization problem that provides the correct optimum sequence of reported measurement data. The remapping algorithm considers a reported measurement in the recovered data

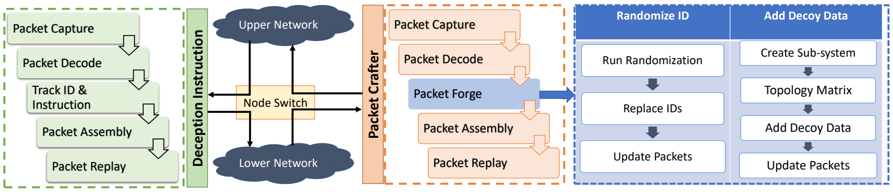


Fig. 3: Deception mechanism in the nodes of the hierarchical network.

only if it is within a specific range of the prediction values. Hence, in the case of stealthy FDI attacks, any sudden change in any measurement reading takes it beyond the allowed threshold; and thus, is not considered in the recovered data. Such removal eliminates the compromised sensors and makes the system immune to the stealthy FDI attacks. However, due to the extensive reduction of compromised sensors or any DoS attack, the remapped measurement data might not be enough to make the system observable. In that case, the missing measurement readings are replaced with the predicted data, which ensures the system's observability.

5 Technical Details

This section elaborates on the working principle of iDDAF.

5.1 Deception Mechanism

Fig. 3 shows the tasks of the nodes for deception mechanism. Firstly, EMS sends the deception instructions to the sensors, and the nodes keep track of the instructions. The instruction contains a type flag where 0, 1, and 2 indicates *fixed*, *randomized*, and *decoyed*, respectively. Thus, each node constrains an m dimensional array $\mathcal{T}$ to store the sensor-wise deception instructions, where m is the number of sensors. Later on, when the sensors report the measurement data, each node in the hierarchy crafts the packets for type 1 and 2 sensors as followings:

```
Algorithm 1: RanID($\mathcal{I}^{rec}, recIDs, randIDs, \mathcal{T}$)
initialize $\mathcal{I}^{rand} = \mathcal{I}^{rec}$;
for $i = 1$ to $len(\mathcal{I}^{rec})$ do
    for $j = 1$ to $len(recID)$ do
        if $\mathcal{I}^{rec}[i] = recID[j]$ and $\mathcal{T}[i] == 1$ then
            $\mathcal{I}^{rand}[i] = randIDs[j]$;
            $break$;
    end
end
```

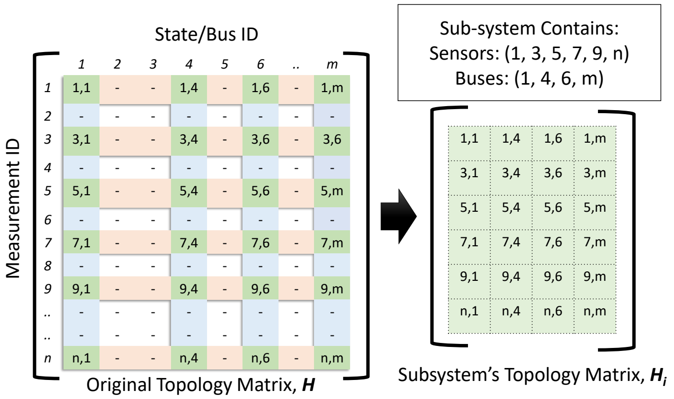
Randomizing IDs The IDs of *randomized* sensors are shuffled among themselves. We propose a heuristic approach in the randomization. Algorithm 1 shows the $RandID$ procedure, which replaces the *randomized* sensors' received IDs $\mathcal{I}^{rec}$ with $\mathcal{I}^{rand}$. The pair ($recIDs$, $randIDs$) contains the lists of IDs that defines randomization pattern for that node. The (*recIDs*, *randIDs*) pairs are generated heuristically and updated at a regular interval by the update instruction from EMS.

As the data packets contain randomized IDs, if the attacker is not aware of the deception, (s)he will be injecting the false data to the deceived locations. Let us assume that the $\mathcal{I}^{org} = \{o_1, o_2, ...o_{m-1}, o_m\}$ is the original sequence of m IDs for the measurement data at one layer, where the shuffled IDs $\mathcal{I}^{rand} = \{r_1, r_2, ...r_{m-1}, r_m\}$ are used with that measurement set. Thus, the probability that $\mathcal{I}^{org}$ and $\mathcal{I}^{rand}$ are exactly the same as $\frac{1}{k!}$, where k is the number of *randomized* sensors and $k \leq m$. For $k = 5$ the probability is 0.008 and $k = 10$ the probability is 2.75×10^{-7}. Thus, for a node with a little higher number of sensors m, such probability converges towards zero. Thus, if an attacker tries to launch a reconnaissance attack, (s)he will end up with different state estimation. On the other hand, if the attacker launches a targeted active attack, (s)he will be attacking the wrong set of sensors. Usually, an FDI attack vector contains a critical set of sensors. Due to this randomization, the attacker attacks sensors with the critical IDs, but they contain the measurement data of sensors coming from different parts of the system. Thus, removing those sensors does not create any issue for the observability of the system.

Algorithm 2: FindDecoy ($\mathbf{z}_{fix}$, $\mathbf{z}_{rand}$, $\mathbf{z}_{dec}$, H, α)

```
initialize the decoy data with the existing data, $\mathbf{z}_{decoy} \leftarrow \mathbf{z}_{dec}$;
$[\mathbf{r}_{fix}, \mathbf{r}_{rand}, \mathbf{r}_{decoy}] = \mathrm{SE}([\mathbf{z}_{fix}, \mathbf{z}_{rand}, \mathbf{z}_{decoy}], H)$;
/* run state estimation and amend decoy data        */
/* repeat until the decoy data is harmonized        */
while $||\mathbf{r}_{decoy}|| > \alpha$ do
    $[\mathbf{r}_{fix}, \mathbf{r}_{rand}, \mathbf{r}_{decoy}] = \mathrm{SE}([\mathbf{z}_{fix}, \mathbf{z}_{rand}, \mathbf{z}_{decoy}], H)$;
    $\mathbf{z}_{decoy} \leftarrow \mathbf{z}_{decoy}$ - $\mathbf{r}_{decoy}$;
end
return $\mathbf{z}_{decoy}$
```

Adding Decoy Data ID randomization makes the deception more visible to the attacker as the random IDs do not follow the topological pattern. Thus, if the adversary runs the **SE-BDD** on the deceptive data that only contains both *randomized* and *fixed* IDs, the sensors with the *randomized* IDs will become outliers, and (s)he might end up with the actual state estimation. Thus, to support the ID randomization, we propose to add decoy data with the (*decoyed*) sensors, which supports the random IDs to be good data in the attacker's state estimation, making the actual (*fixed*) data outliers. Let's assume that $\mathbf{z}^i$ ($\in \mathbf{z}$) is the measurement vector consisting of only the sensors of i-th node. We define H_i as the topology matrix of the i-th sub-system, where the rows represent the sensors in $\mathbf{z}^i$, and the columns are for the substations where those sensors are located. Fig. 4 shows a case of generating H_i from the system's full topology matrix H. Thus, we can define $\mathbf{z}^i$ as $[\mathbf{z}^i_{fix}, \mathbf{z}^i_{rand}, \mathbf{z}^i_{dec}]$, where $\mathbf{z}^i_{fix}$, $\mathbf{z}^i_{rand}$, and $\mathbf{z}^i_{dec}$ contains the fixed, randomized, and decoyed measurement data. Algorithm 2


State/Bus ID
Measurement ID
Original Topology Matrix, *H*
Sub-system Contains:
Sensors: (1, 3, 5, 7, 9, n)
Buses: (1, 4, 6, m)
Subsystem's Topology Matrix, ***H****ᵢ*

	1	2	3	4	5	6	..	m
1	1,1	-	-	1,4	-	1,6	-	1,m
2	-	-	-	-	-	-	-	-
3	3,1	-	-	3,4	-	3,6	-	3,6
4	-	-	-	-	-	-	-	-
5	5,1	-	-	5,4	-	5,6	-	5,m
6	-	-	-	-	-	-	-	-
7	7,1	-	-	7,4	-	7,6	-	7,m
8	-	-	-	-	-	-	-	-
9	9,1	-	-	9,4	-	9,6	-	9,m
..	-	-	-	-	-	-	-	-
..	-	-	-	-	-	-	-	-
n	n,1	-	-	n,4	-	n,6	-	n,m

1,1	1,4	1,6	1,m
3,1	3,4	3,6	3,m
5,1	5,4	5,6	5,m
7,1	7,4	7,6	7,m
9,1	9,4	9,6	9,m
n,1	n,4	n,6	n,m

Fig. 4: Generation of sub-system topology matrix H_i.

shows the technique to calculate the decoy data $\mathbf{z}^i_{dec}$. The procedure takes the sub-system's randomized data, topology matrix H^i, and a threshold α.

5.2 Remapping Mechanism

This section introduces the prediction-based remapping. Fig. 5 shows the process of remapping using a second order polynomial regression model. In the following subsections, we explain different modules of the prediction-based remapping.

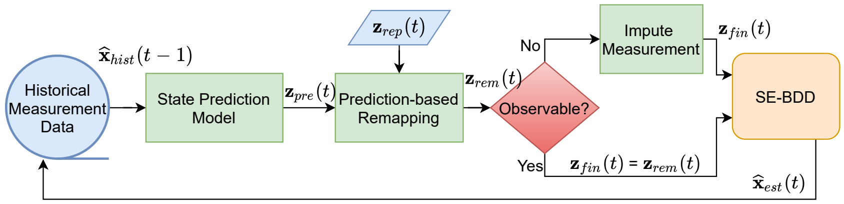

Historical Measurement Data
$\hat{\mathbf{x}}_{hist}(t-1)$
State Prediction Model
$\mathbf{z}_{pre}(t)$
$\mathbf{z}_{rep}(t)$
Prediction-based Remapping
$\mathbf{z}_{rem}(t)$
Observable?
No
Yes
Impute Measurement
$\mathbf{z}_{fin}(t)$
$\mathbf{z}_{fin}(t) = \mathbf{z}_{rem}(t)$
SE-BDD
$\hat{\mathbf{x}}_{est}(t)$

Fig. 5: Prediction-based remapping mechanism of iDDAF.

Sensor Prediction The first part of remapping is to train the regression model based on the past state estimation data, which estimates the current state vector. The estimated state vector is multiplied with a topology matrix H to generate the predicted measurement vector. The prediction model takes past k samples of the previous estimated state vector, $\hat{\mathbf{x}}^{hist}(t-1)$ and predict the next state $\mathbf{x}^{pre}(t)$, which generates the measurement sample, $\mathbf{z}^{pre}(t) = H\ \mathbf{x}_{pre}(t)$. Hence, in the regression algorithm, we provide an $n \times k$ dimensional historical state data $\hat{\mathbf{x}}_{hist}(t-1)$ and generate $n \times 1$ dimensional $\mathbf{x}^{pre}(t)$. Here, $\hat{\mathbf{x}}_{hist}(t-1) = [\hat{\mathbf{x}}(t-1)$, $\hat{\mathbf{x}}(t-2)$, ..., $\hat{\mathbf{x}}(t-k+1)$, $\hat{\mathbf{x}}(t-k)]$.

Sensor Remapping Due to the deception, the sensor's measurement is crafted with randomized IDs, and the reported measurement vector, $\mathbf{z}_{rep}(t)$, needs to be reshuffled to get back to the original pattern. To do this remapping, we

Table 1: Modeling Parameters of Remapping Algorithm.

Notation	**Type, Dimension**	**Definition**
$\mathbf{z}_{pre}$	1-D Array, $n_p \times 1$	Set of predicted measurements
n_p	Integer	Num of predicted measurements
$\mathbf{z}_{rep}$	1-D Array, $n_s \times 1$	Set of reported measurements
n_r	Integer	Num of reported measurements
$\mathcal{M}$	2-D Array, $n_p \times n_s$	Recovery mapping matrix
$\mathcal{F}$	2-D Array, $n_p \times n_s$	Fixed sensor mapping matrix
$\mathcal{D}$	1-D Array, $n_r \times 1$	Decoy mapping array
$\mathcal{C}$	2-D Array, $n_p \times n_s$	Recovery cost matrix
$\mathcal{O}_{Reco}^{Cost}$	Integer	Measurement recovery cost
$\mathcal{O}_{Assi}^{Prof}$	Integer	Measurement assignment profit
η	Integer	Recovery threshold

design an optimization algorithm to assign the best set of sensors IDs to $\mathbf{z}_{rem}(t)$ considering the predicted data $\mathbf{z}_{pre}(t)$.

Remapping algorithm: This part explains the repairing algorithm. As we deal with different combinations of the data, we use mixed integer programming (MIP) to implement the combinatorial optimization problem. Table 1 shows the notations used in defining the constraints. Our goal is to find the $(n_p \times n_r)$ dimensional binary matrix $\mathcal{M}$, where the positions of the ones represent the successful recovery of reported measurements. The rows and columns of the ones in $\mathcal{M}$ represent the IDs of predicted and reported measurement, respectively. For example, a one in the position (i, j) of $\mathcal{M}$ represents that the j-th reported reading in $\mathbf{z}_{rep}$ is the actual measurement of i-th sensor. Thus, the total number of ones in $\mathcal{M}$ indicates the number of recovered measurements.

fixed sensors do not participate in the randomization process during deception. Their IDs remain the same during the whole process and we explicitly define them during the repairing process. $\mathcal{F}$ is the 2-D binary mapping matrix, where the rows and columns represent the already known remapping of *fixed* sensors. On the other hand, $\mathcal{D}$ presents the 1-D binary mapping of the *decoyed* sensors. As shown in (2) and (3), each recovery of the randomized sensor comes with a cost, defines as the difference between the predicted and the reported value. In the case of a fixed ID, the cost is explicitly defined as zero. As shown in (4), the *decoyed* measurements are not assigned to any of the sensors.

$$\forall_{1,1 \leq i,j \leq n_s,n_r} \;\; \mathcal{F}_{i,j} == 0 \implies \mathcal{C}_{i,j} == |\mathbf{z}_{pre}^{i} - \mathbf{z}_{rep}^{j}| \tag{2}$$

$$\forall_{1,1 \leq i,j \leq n_s,n_r} \;\; \mathcal{F}_{i,j} == 1 \implies (\mathcal{M}_{i,j} == 1) \;\wedge\; (\mathcal{C}_{i,j} == 0) \tag{3}$$

$$\forall_{1 \leq j \leq n_r} \;\; \mathcal{D}_j == 1 \implies \sum_{i=1}^{n_s} \; \mathcal{M}_{i,j} == 0 \tag{4}$$

The recovery data must be within a specific range of the predicted data. Thus, as shown in (5), a recovery is valid only if the associated cost is within $\eta\%$ of the expected data. However, if there is no such reported value within that range, that sensor remains unassigned.

$$\forall_{1\leq i,j\leq n_s,n_r} \ \mathcal{C}_{i,j} > |\ \mathbf{z}^{i}_{pre} \times \eta| \implies \mathcal{M}_{i,j} == 0 \tag{5}$$

The constraints in (6) mandate that any measurement can be assigned to atmost one sensor and vice-versa.

$$\forall_{1\leq i\leq n_s} \sum_{j=1}^{n_r} \ \mathcal{M}_{i,j} \leq 1 \quad \text{and} \quad \forall_{1\leq j\leq n_r} \sum_{i=1}^{n_s} \ \mathcal{M}_{i,j} \leq 1 \tag{6}$$

As shown in (7), the ultimate goal is to assign as much measurements as possible (maximize assignment profit), while keeping the recovery cost to minimum. This two objective functions are merged together in (8). Thus, the optimization maximizes the assignment profit by allocating as many sensors as possible and minimizes the recovery cost by assigning to the closest prediction. To ensure the maximum number of recovered sensors, we multiply the assignment profit by k and emphasize it than the recovery cost. Hence, there will always be a solution; however, no sensor will be recovered in the worst-case scenario. EMS will rely only on the fixed sensors to run the state estimation in such a rare case. Thus, it will be useful to select a set of fixed sensors that spans the system's critical parts and ensures observability.

$$\mathcal{O}^{Cost}_{Reco} = \sum_{i=1}^{n_s}\sum_{j=1}^{n_r} \ \mathcal{M}_{i,j} \times \mathcal{C}_{i,j} \qquad and \quad \mathcal{O}^{Prof}_{Assi} = \sum_{i=1}^{n_s}\sum_{j=1}^{n_r} \ \mathcal{M}_{i,j} \tag{7}$$

$$\min \ \ (\mathcal{O}^{Cost}_{Reco} - k \times \mathcal{O}^{Prof}_{Assi}) \tag{8}$$

The successful execution of the program returns the matrix $\mathcal{M}$ from where we find the remapped pattern $\mathbf{z}_{rem} = \mathcal{M} \times \mathbf{z}_{rep}$, which is finally used by EMS in $SE - BDD$. If the system is observable, it finds the state vectors and takes the necessary control decision. However, if the system is unobservable, we use an imputation algorithm to fill up the missing data.

Data Imputation: If there is an L-1/L-2 FDI attack in the system, the recovery algorithm can remove the compromised measurement and may keep the system observable under sufficient randomization. However, in the case of L-0 attacks, dropping the compromised measurement during the repairing process may lead the system to unobservability. Besides, instead of an FDI attack, there can be a DoS attack, where the attacker's goal is to make some targeted critical measurement missing. In both cases, iDDAF replaces the missing data with the predicted measurements from $\mathbf{z}_{pre}(t)$.

5.3 A 14 Bus Case Study

In this section, we provide a case study of prediction-based remapping in the IEEE 14 bus system [3] as shown Fig. 6. For the simplicity, we consider a case where 100% of sensors are reporting measurement data to EMS, and all of them are considered for randomization.

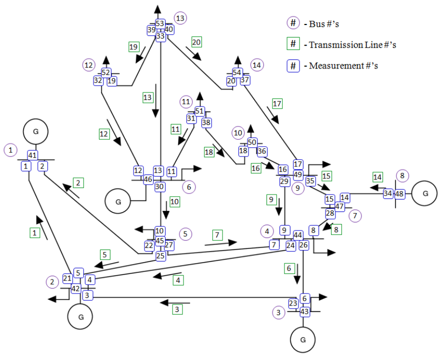
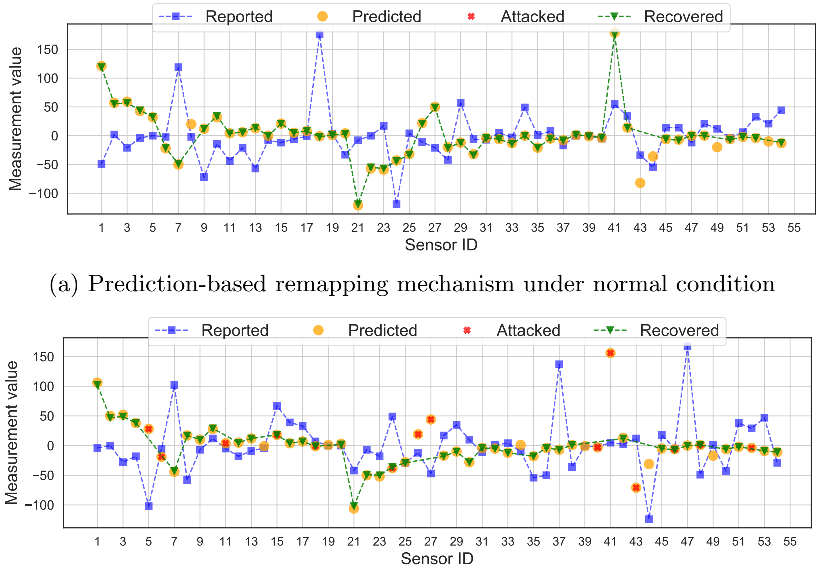


Fig. 6: IEEE 14-bus test system [28]

Remapping under normal condition Fig. 7a shows the randomized reported measurement data, sensor-wise prediction, and the recovered data under normal operating conditions. The reported data contain random IDs; thus, the reported measurement vector's shape does not follow the predicted points. However, once the remapping algorithm assigns the random measurement values to the right IDs, the recovered data precisely follow the predicted data. However, sensors 8, 43, 44, 49, and 53 are not assigned to any measurement as the model cannot find any possible candidate for them. Those sensors may contain noises that deviate them from the predicted values. However, among 54 measurements, 49 of them are assigned to the right IDs, which is enough to make the system observable.



(a) Prediction-based remapping mechanism under normal condition

(b) Prediction-based remapping mechanism under FDI attack

Fig. 7: Prediction-based remapping under normal and attack conditions.

Remapping under FDI attack condition Fig. 7b shows how prediction-based remapping eliminates the impact of the FDI attacks. In this case, we consider an FDI attack, where the targeted sensors are 8, 9, 15, 16, 17, 28, 29, 35, 36, 37, 44, 47, 50, and 54. However, due to the deception, these sensor IDs are used to send the data of 5, 6, 11, 15, 18, 24, 26, 27, 40, 41, 43, 46, 48, and

52, respectively. Hence, even though the attacker expects to be stealthy and bypass the BDD, the remapping mechanism mitigates the attack by declining the compromised sensors. Thus, among the attacked sensors 5, 6, 11, 26, 27, 40, 41, 43, and 52 are unassigned due to their suspicious values, which alleviates the impact of the FDI attack. Sensors 15, 18, 24, and 48 are not removed as their injection amounts are very small, within 5% of the predicted values. Further processing on this measurement data in the state estimation process removes the remaining outliers and make the system effectively immune to the attack.

6 Evaluation

We evaluate the robustness of iDDAF against three different types of attacks (e.g., Reconnaissance, FDI, and DoS) each in three different levels (e.g., L-0, L-1, and L-2). We run the evaluation on IEEE 14 bus system considering the attacker can compromise the maximum of five buses at a time. We use the 365 days of synthetic IEEE 14 bus system time-series data [1] to evaluate the effectiveness of the ID randomization and data modification.

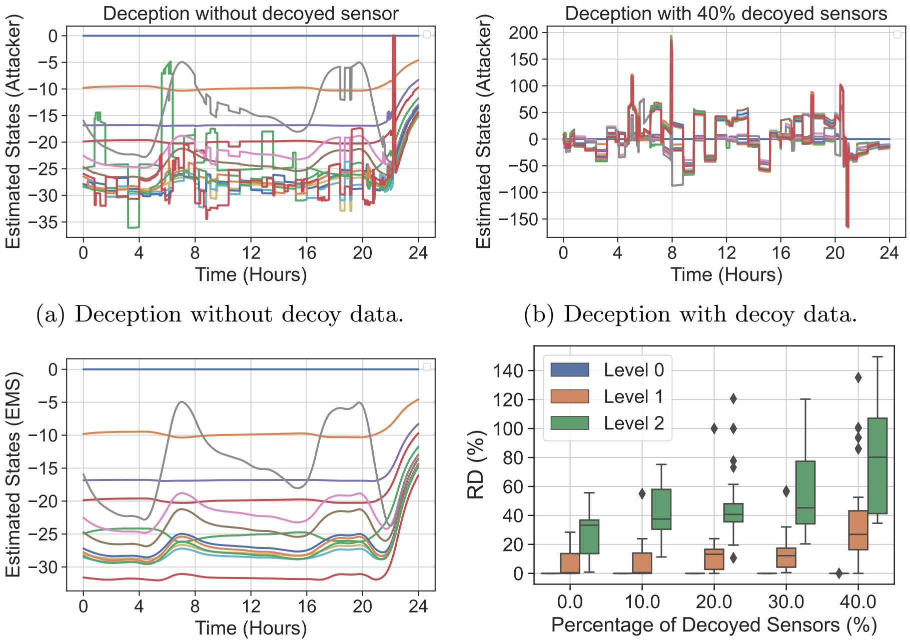
We consider the intruder to be in the system for the entire evaluation period during a reconnaissance attack. On the other hand, we initiate the active attacks in every 30 time steps, continue for different durations, and observe whether the state estimation deviates during the executions.

6.1 Evaluation metrics

The following metrics are used to evaluate iDDAF's performance:
Reconnaissance Deviation (RD) is defined as the percentage of the deviation between the actual and attacker's estimated states. A higher RD indicates the iDDAF's success in misleading the reconnaissance attacks.
Estimation Deviation (ED) is defined as the percentage of deviation between the actual and EMS's estimated states. Unlike RD, a lower ED indicates the iDDAF's success in recovering from the active attacks. Thus, $RD = \frac{||\hat{x}_{act} - \hat{x}_{dec}||}{||\hat{x}_{act}||} \times 100$, and $ED = \frac{||\hat{x}_{act} - \hat{x}_{ems}||}{||\hat{x}_{act}||} \times 100$, where $\hat{x}_{act}$, $\hat{x}_{dec}$, and $\hat{x}_{ems}$, are the actual, attacker's, and EMS's estimated state vector.
Percentage of Unobservable Cases (PUC) is defined as the percentage of attacks that create unobservability in the defender's estimation. The primary goal of DoS attacks is to increase the PUC as much as possible.

6.2 Reconnaissance Attack

This part shows how iDDAF ensures the system's privacy by randomizing the IDs and then adding decoy data. In this evaluation, we consider 60% of the sensors as *randomized*. With the remaining 40% sensors, we study the contribution of the decoy data and observe how the attacker's state estimation deviates from the actual ones. First, we consider the remaining 40% as fixed sensors (without adding any decoy data). In such a case, Fig. 8a shows an L-2 attacker is able to estimate the trend of the actual system states, even without 60% *randomized*



(a) Deception without decoy data.

(b) Deception with decoy data.

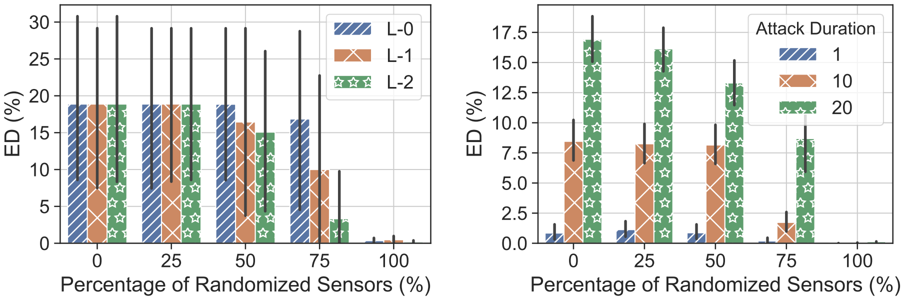
(c) EMS's estimated states.

(d) Impact of decoy data on RD.

Fig. 8: iDDAF's performance against reconnaissance attacks. (a-c: different colors indicate different states of different substations).

sensors. As most of the measurements with the random IDs become bad data, the attacker is left with most of the fixed sensors as good data as they are topologically aligned. Later, we consider that 40% of the *fixed* sensors as *decoyed* sensors and added decoy data to validate the ID randomization. In this case, the random IDs get the support and become good data in the attacker's estimation and do not become the outliers. Fig. 8b shows the attacker's estimation deviates from the actual ones and moves toward random directions whenever the nodes update the randomization patterns. On the other hand, Fig. 8c illustrates, EMS successfully remaps the IDs and estimates the state accurately. Even after 24 hours of remapping, there is no deviation/overshooting from the actual values, which indicates the robustness of iDDAF.

Fig. 8d shows RD with 60% *randomized* sensors and different *decoyed* sensors for the passive attackers at different levels. As decoy data are added at the switch-levels (L-1/L-2), not at the sensors levels (L-0), RD is zero in case of L-0 attacks. Such attacks are costly and infeasible as the attacker needs to be distributed and compromise the targeted sensors locally. A practical approach for reconnaissance is to compromise the network devices (L-1/L-2). In that case, adding more *decoyed* sensors supports the randomization, increases RD, and thus, misleads the attacker. Although iDDAF is ineffective against L-0 passive attacks, the following analysis shows, it can completely mitigate the L-0 active attacks, making such passive attacks futile.



(a) FDI attacks on different levels (b) FDI attacks with different duration.

Fig. 9: iDDAF's performance against FDI attacks.

6.3 FDI Attack

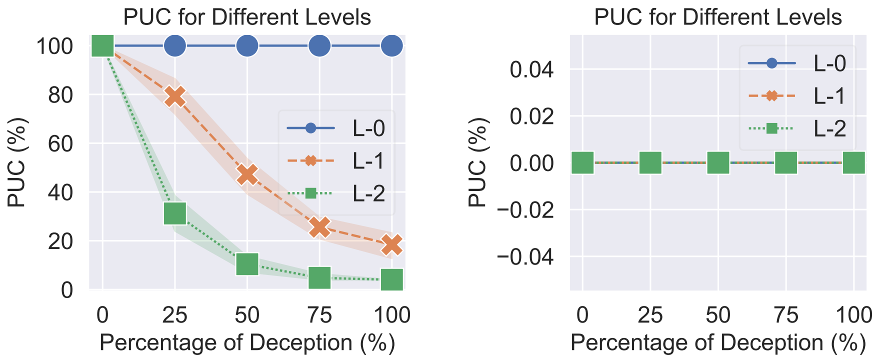
Single-step attack: Fig. 9a shows the system's robustness under single step FDI attacks in different levels and randomization. In this case, we vary the percentage of *randomized* sensors from 0% to 100% and consider the remaining as *fixed*. Here, the case of 0% *randomized* (100% *fixed*) sensors represents a system without any deception defense. Thus, all the FDI attacks compromise the correct sensors and remain stealthy with an ED of 20%. However, as we add more *randomized* sensors, the injections start to happen in the wrong places. As the *randomized* sensors also go through the remapping (filtering) algorithm, the malicious data are filtered out if they do not follow the predicted trend; and thus, the attack impacts are alleviated irrespective of the levels. With 100% *randomized* sensors, almost all the compromised measurements are removed by remapping algorithm and the attack impacts are completely mitigated. Although such filtering in L-0 FDI attacks may lead the system to unobservable situation, the critical missing data are imputed using the predicted values; and thus, the system retains the observability.

Multi-steps attack: In this part, we analyze the impact of continuous FDI attacks on that state estimation. We consider the case of 1, 10, and 20 time-steps as the attack duration. In each case, a random L-2 FDI attack is initiated at every 30 time steps and continued for the considered duration. Fig. 9b shows that ED for long-duration attack is higher due to the cascading effect. However, as more sensors are considered in randomization, the initial attack impact is perfectly handled by the remapping and estimation process so that no cascading effect is observed, making iDDAF robust against influential continuous attacks.

6.4 DoS Attack

This part evaluates iDDAF's defense against targeted DoS attacks. We consider the DoS attacks compromising the same sensors as we do in FDI attacks. However, instead of injecting false data, the attacker drops the packets of the targeted critical set of sensors to make the system unobservable.

Sharing-based remapping: First, we show the performance of existing sharing-based remapping mechanisms, where the actual remapping pattern is already



(a) Sharing-based remapping.

(b) iDDAF with data imputation.

Fig. 10: iDDAF's performance against DoS attacks.

shared with the EMS. Fig. 10a shows the PUC under DoS attacks at different levels. In the case of L-0 DoS attacks, sharing-based remapping completely fails to ensure the system's observability, as the attacker directly compromises the sensors. The randomization in the nodes after the L-0 DoS attack is not helpful anymore. However, in the case of L-1 or L-2 attacks, even though randomization deceives the attacker, in some cases, the system may still lose observability.
iDDAF's predictive remapping: On the other hand, Fig. 10b shows the same attack scenarios but with remapped with iDDAF. The figure shows iDDAF is entirely immune to any level of DoS attacks. Even in the worst-case scenario of L-0 DoS attacks, the missing critical measurements are replaced using the predicted values, and the system recovers from the DoS attacks.

7 Related Work

To deal with different cyber-attacks and mitigate sensory channels FDI attacks, several moving target defense (MTD) techniques are proposed. Several works introduced uncertainty/randomness into the CPS control loop. Griffith et al. proposed a method that includes stochastic and time-varying parameters in the control loops for CPSs [11]. To detect and minimize the impact of FDI attacks, Giraldo presented MTD by randomly varying the availability of telemetry sensor data [10]. Rahman et al. reduced the attack window by adding an uncertainty factor to the subset of sensors used to estimate the system's state [30]. Based on the skewness coefficients, Hu et al. proposed a stealthy attack detection strategy capable of identifying the forged residuals from the attack-free residuals [15]. Additionally, some authors examined methods to disrupt the physical properties of the system to invalidate its attack vectors. Lakshminarayana et al. proposed a formal model for reactance perturbation based MTD using D-FACTS [18]. Tian et al. proposed a hidden MTD using D-FACTs in smart grids to defend structured FDI attacks [36].

Several research studies focused on confounding the attacker using crafty network packets. In [21], Li et al. proposed CPSMorph to create several fake network sessions and the actual ones to hide them from attackers. Pappa et al.

proposed an end-to-end IP hoping in the CPS SCADA system that uses seed value to share the randomization information [27]. The seeds were used to generate random IP addresses, which they shared over a public-private encryption channel. In [12], Groat et al. proposed a secured IPv6-based smart grid communication system titled, MT6D [12]. They implemented security at the network layer to defend against most of the IP-specific attacks.

In addition, some other research works are carried out on CPS for deceptive defense against stealthy attacks. Lin et al. proposed a randomized data acquisition into multiple rounds [22]. An SDN-enabled framework controls the flows in the network and collects measurements from randomly selected online sensors while spoofing data from the remainder. However, a clever attacker can inject false data into online devices since only a few sensors send original data at a given time. An attacker may also be able to identify the correct measurement by examining the pattern of sensor measurements. Additionally, they proposed virtualizing physical functions and crafting decoy data to disrupt reconnaissance attacks on power grids [23]. An attacker can still inject the FDI attack into a specific part of the system with no virtual nodes while ignoring the rest. Consequently, their proposed solution only secures the parts of the systems where virtual nodes are located, leaving other parts unsecured.

In the works mentioned above [22, 23], the data acquisition process is partially randomized, and only the mitigation of attacks is addressed. The existing solutions are attack-specific and thus, do not provide complete immunity. In contrast, our proposed iDDAF can be implemented on the entire system leaving no window for the attacker. Furthermore, iDDAF is the only framework that provides a complete security solution (CIA) against three different types of attacks.

8 Conclusion

CPSs are omnipresent in modern critical infrastructures. Moreover, the extensive integration of IoT technologies converts the CPSs into smart, efficient, and complex hybrid systems. However, such dependency creates a vast attack space that the attackers can exploit. Thus, to secure the CPSs hierarchical networks, we present iDDAF, an intelligent and secure data acquisition framework. iDDAF introduces deception while collecting measurement data by randomizing the sensor IDs and adding decoy data. The control center utilizes an a prediction model to predicts future measurement. Utilizing the expected data, an optimization algorithm recovers the original data sequence from the reported random/decoy data. The evaluation results on the standard IEEE 14 bus system demonstrate that iDDAF can successfully mitigate different influential cyberattacks through intelligent deception and remapping steps.

9 Acknowledgement

This research was partially supported by National Science Foundation (NSF) under award #1929183.

References

1. iddaf. https://sites.google.com/view/iddaf/home
2. Iec 61850-power utility automation. https://www.iec.ch/smartgrid/standards/
3. Ieee 14-bus system. https://www.icseg.iti.illinois.edu/ieee-14-bus-system/
4. Abur, A., Exposito, A.G.: Power system state estimation: theory and implementation. CRC press (2004)
5. Ali, S., Qaisar, S.B., Saeed, H., Khan, M.F., Naeem, M., Anpalagan, A.: Network challenges for cyber physical systems with tiny wireless devices: A case study on reliable pipeline condition monitoring. Sensors (4), 7172–7205 (2015)
6. Barua, A., Al Faruque, M.A.: Hall spoofing: A non-invasive dos attack on grid-tied solar inverter. In: 29th {USENIX} Security Symposium ({USENIX} Security 20)
7. Doherty, J.: SDN and NFV simplified: a visual guide to understanding software defined networks and network function virtualization. Addison-Wesley Professional
8. Dorsch, N., Kurtz, F., Georg, H., Hägerling, C., Wietfeld, C.: Software-defined networking for smart grid communications: Applications, challenges and advantages. In: 2014 IEEE international conference on smart grid communications
9. Drinkwater, D.: Stuxnet-style attack. http://www.innotap.com/2015/07/stuxnet-style-attack-on-us-smart-grid-could-cost-government-1-trillion/
10. Giraldo, J., C., A., S., R.G.: A moving target defense to detect stealthy attacks in cyber-physical systems. In: 2019 American Control Conference. IEEE
11. Griffioen, P., Weerakkody, S., Sinopoli, B.: A moving target defense for securing cyber-physical systems. IEEE Transactions on Automatic Control (2020)
12. Groat, S., Dunlop, M., Urbanksi, W., Marchany, R., Tront, J.: Using an ipv6 moving target defense to protect the smart grid. In: 2012 IEEE PES Innovative Smart Grid Technologies (ISGT). pp. 1–7. IEEE (2012)
13. Gunes, V., Peter, S., Givargis, T., Vahid, F.: A survey on concepts, applications, and challenges in cyber-physical systems. KSII Transactions on Internet & Information Systems **8**(12) (2014)
14. Haque, N.I., Shahriar, M.H., Dastgir, M.G., Debnath, A., Parvez, I., Sarwat, A., Rahman, M.A.: Machine learning in generation, detection, and mitigation of cyberattacks in smart grid: A survey. arXiv preprint arXiv:2010.00661 (2020)
15. Hu, Y., Li, H., Yang, H., Sun, Y., Sun, L., Wang, Z.: Detecting stealthy attacks against industrial control systems based on residual skewness analysis. EURASIP Journal on Wireless Communications and Networking (1), 74 (2019)
16. Jafari, M., Shahriar, M.H., Rahman, M.A., Paudyal, S.: False relay operation attacks in power systems with high renewables. arXiv preprint arXiv:2102.12041
17. Kovacs, E.: Blackenergy malware used in ukraine power grid attacks (2016)
18. Lakshminarayana, S., Yau, D.K.: Cost-benefit analysis of moving-target defense in power grids. In: 2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN). pp. 139–150. IEEE (2018)
19. Langner, R.: Stuxnet: Dissecting a cyberwarfare weapon. IEEE Security & Privacy
20. Li, B., Lu, R., Wang, W., Choo, K.K.R.: Ddoa: A dirichlet-based detection scheme for opportunistic attacks in smart grid cyber-physical system. IEEE Transactions on Information Forensics and Security (11), 2415–2425 (2016)
21. Li, Y., Dai, R., Zhang, J.: Morphing communications of cyber-physical systems towards moving-target defense. In: 2014 IEEE International Conference on Communications (ICC). pp. 592–598. IEEE (2014)

22. Lin, H., Kalbarczyk, Z.T., Iyer, R.K.: Raincoat: Randomization of network communication in power grid cyber infrastructure to mislead attackers. IEEE Transactions on Smart Grid (5), 4893–4906 (2018)
23. Lin, H., Zhuang, J., Hu, Y.C., Zhou, H.: Defrec: Establishing physical function virtualization to disrupt reconnaissance of power grids' cyber-physical infrastructures. In: The Proceedings of 2020 Network and Distributed System Security Symposium
24. Liu, Y., Ning, P., Reiter, M.K.: False data injection attacks against state estimation in electric power grids. ACM Transactions on Information and System Security
25. Lu, Z., Sun, C., Cheng, J., Li, Y., Li, Y., Wen, X.: Sdn-enabled communication network framework for energy internet. Journal of Computer Networks and Communications
26. Newaz, A., Sikder, A.K., Rahman, M.A., Uluagac, A.S.: A survey on security and privacy issues in modern healthcare systems: Attacks and defenses
27. Pappa, A.C., Ashok, A., Govindarasu, M.: Moving target defense for securing smart grid communications: Architecture, implementation & evaluation. In: 2017 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)
28. Rahman, M.A., Al-Shaer, E., Kavasseri, R.: Impact analysis of topology poisoning attacks on economic operation of the smart power grid. In: International Conference on Distributed Computing Systems (ICDCS) (Jul 2014)
29. Rahman, M.A., Shahriar, M.H., Masum, R.: False data injection attacks against contingency analysis in power grids: poster. In: Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks. pp. 343–344 (2019)
30. Rahman, M.A., Al-Shaer, E., Bobba, R.B.: Moving target defense for hardening the security of the power system state estimation. In: Proceedings of the First ACM Workshop on Moving Target Defense. pp. 59–68 (2014)
31. Rahman, M.A., Shahriar, M.H., Jafari, M., Masum, R.: Novel attacks against contingency analysis in power grids. arXiv preprint arXiv:1911.00928 (2019)
32. Shahriar, M.H., Haque, N.I., Rahman, M.A., Alonso, M.: G-ids: Generative adversarial networks assisted intrusion detection system. In: IEEE 44th Annual Computers, Software, and Applications Conference (COMPSAC). IEEE (2020)
33. Shahriar, M.H., Sadiq, M.J., Uddin, M.F.: Stability analysis of grid connected pv array under maximum power point tracking. In: 2016 9th International Conference on Electrical and Computer Engineering (ICECE). pp. 499–502. IEEE (2016)
34. Sikder, A.K., Petracca, G., Aksu, H., Jaeger, T., Uluagac, A.S.: A survey on sensor-based threats and attacks to smart devices and applications. IEEE Communications Surveys & Tutorials (2021)
35. Simmonds, A., Sandilands, P., Van Ekert, L.: An ontology for network security attacks. In: Asian Applied Computing Conference. pp. 317–323. Springer (2004)
36. Tian, J., Tan, R., Guan, X., Liu, T.: Enhanced hidden moving target defense in smart grids. IEEE Transactions on Smart Grid (2), 2208–2223 (2018)
37. Tzu, S., Tzu, S., Sun, W., Vu, S.C., et al.: The art of war, vol. 361. Oxford University Press, USA (1971)
38. Van Hertem, D., Verboomen, J., Purchala, K., Belmans, R., Kling, W.L.: Usefulness of dc power flow for active power flow analysis with flow controlling devices. In: The 8th IEE International Conference on AC and DC Power Transmission. pp. 58–62 (March 2006). https://doi.org/10.1049/cp:20060013
39. Wang, S., Zhang, Y., Yang, Z., Chen, Y.: A graphical hierarchical cps architecture. In: 2016 International Symposium on System and Software Reliability. IEEE
40. Yampolskiy, M., Horvath, P., Koutsoukos, X.D., Xue, Y., Sztipanovits, J.: Systematic analysis of cyber-attacks on cps-evaluating applicability of dfd-based approach. In: 2012 5th International Symposium on Resilient Control Systems. IEEE