Masthead Logo University of Tennessee, Knoxville
Trace: Tennessee Research and Creative Exchange

Doctoral Dissertations Graduate School

12-2018

Vulnerability Assessment and Privacy-preserving Computations in Smart Grid

Xiangyu Niu
University of Tennessee, xniu@vols.utk.edu

To the Graduate Council:

I am submitting herewith a dissertation written by Xiangyu Niu entitled "Vulnerability Assessment and Privacy-preserving Computations in Smart Grid." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Computer Engineering.

Jinyuan Sun, Major Professor

We have read this dissertation and recommend its acceptance:

Xueping Li, Hairong Qi, Kevin Tomsovic

Accepted for the Council:
Carolyn R. Hodges

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Vulnerability Assessment and Privacy-preserving Computations in Smart Grid

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Xiangyu Niu

December 2018

imagine

Acknowledgments

I would like first to express my special thanks to Dr. Jinyuan Sun, for the patient guidance, encouragement, and advice she has provided throughout my graduate study. Her constant support and wonderful personality are essential to the completion of this dissertation.

My gratitude also goes to the members of my committee: Dr. Kevin Tomsovic, Dr. Hairong Qi, and Dr. Xueping Li whose helpful comments and suggestions were an enormous help to me. I also would like to thank my colleagues and friends for supporting my research and sharing their life together. Also, I am very thankful to CURENT for the financial support and for providing me with a fantastic atmosphere to conduct my research.

Last but not least, I would like to express my appreciation to my family for their moral support and warm encouragements.

Abstract

Modern advances in sensor, computing, and communication technologies enable various smart grid applications which highlight the vulnerability that requires novel approaches to the field of cybersecurity. While substantial numbers of technologies have been adopted to protect cyber attacks in smart grid, there lacks a comprehensive review of the implementations, impacts, and solutions of cyber attacks specific to the smart grid.

In this dissertation, we are motivated to evaluate the security requirements for the smart grid which include three main properties: confidentiality, integrity, and availability. First, we review the cyber-physical security of the synchrophasor network, which highlights all three aspects of security issues. Taking the synchrophasor network as an example, we give an overview of how to attack a smart grid network. We test three types of attacks and show the impact of each attack consisting of denial-of-service attack, sniffing attack, and false data injection attack.

Next, we discuss how to protect against each attack. For protecting availability, we examine possible defense strategies for the associated vulnerabilities.

For protecting data integrity, a small-scale prototype of secure synchrophasor network is presented with different cryptosystems. Besides, a deep learning based time-series anomaly detector is proposed to detect injected measurement. Our approach observes both data measurements and network traffic features to jointly learn system states and can detect attacks when state vector estimator fails.

For protecting data confidentiality, we propose privacy-preserving algorithms for two important smart grid applications. 1) A distributed privacy-preserving quadratic optimization algorithm to solve Security Constrained Optimal Power Flow (SCOPF) problem. The SCOPF problem is decomposed into small subproblems using the Alternating Direction

Method of Multipliers (ADMM) and gradient projection algorithms. 2) We use Paillier cryptosystem to secure the computation of the power system dynamic simulation. The IEEE 3-Machine 9-Bus System is used to implement and demonstrate the proposed scheme. The security and performance analysis of our implementations demonstrate that our algorithms can prevent chosen-ciphertext attacks at a reasonable cost.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Background

A smart grid uses information and control technologies to improve reliability, security, and efficiency of the traditional electric power system. Many countries have already started extensive research and development on smart grid technologies. A report by Pike Research [101] states that the European Union's investment in smart grid development will increase to $79 billion by 2020 . The U.S. also estimated to contribute ranging from $23.8 billion to $44 billion annually to the smart grid deployment by 2030 [2].

According to EPRI's study [44], the potential benefits of the smart grid are as follows:

- Allows Direct Participation by Consumers. The smart grid consumer is informed, modifying the way they use and purchase electricity. They have choices, incentives, and disincentives.

- Accommodates all Generation and Storage Options. The smart grid accommodates all generation and storage options.

- Enables New Products, Services, and Markets. The smart grid enables a market system that provides cost-benefit tradeoffs to consumers by creating opportunities to bid for competing services.

- Provides Power Quality for the Digital Economy. The smart grid provides reliable power that is relatively interruption-free.

- Optimizes Asset Utilization and Operational Efficiently. The smart grid optimizes assets and operates efficiently.

- Anticipates and Responds to System Disturbances (Self-heal). The smart grid independently identifies and reacts to system disturbances and performs mitigation efforts to correct them.

- Operates Resiliently against Attack and Natural Disaster. The smart grid resists attacks on both the physical infrastructure (substations, poles, transformers, etc.) and the cyber-structure (markets, systems, software, communications).

To achieve the aforementioned benefits, National Institute of Standards and Technology (NIST) under the terms of the 2007 Energy Independence and Security Act (EISA) is responsible for coordinating the development of interoperability standards that support the implementation of the smart grid. Based on the latest NIST guideline [50], a specific conceptual diagram with seven domains for smart grid is depicted in Figure 1.1 which groups the smart grid based on objectives, devices and functions that have similar applications. This work primarily focus on the information network in this diagram which contains interconnected computers, communication devices, and additional information and communication technologies that share information across different entities to enable the automatic supply and demand balance and real-time monitoring. More specifically, we are interested in finding the vulnerabilities that exist in the information network of smart grid and carrying out novel solutions to fill in the gaps where existing cybersecurity schemes do not work.

1.2 Motivation

The next-generation power grid transforms from a centralized network to a more decentralized and consumer-interactive network. This transformation is enabled by two-way

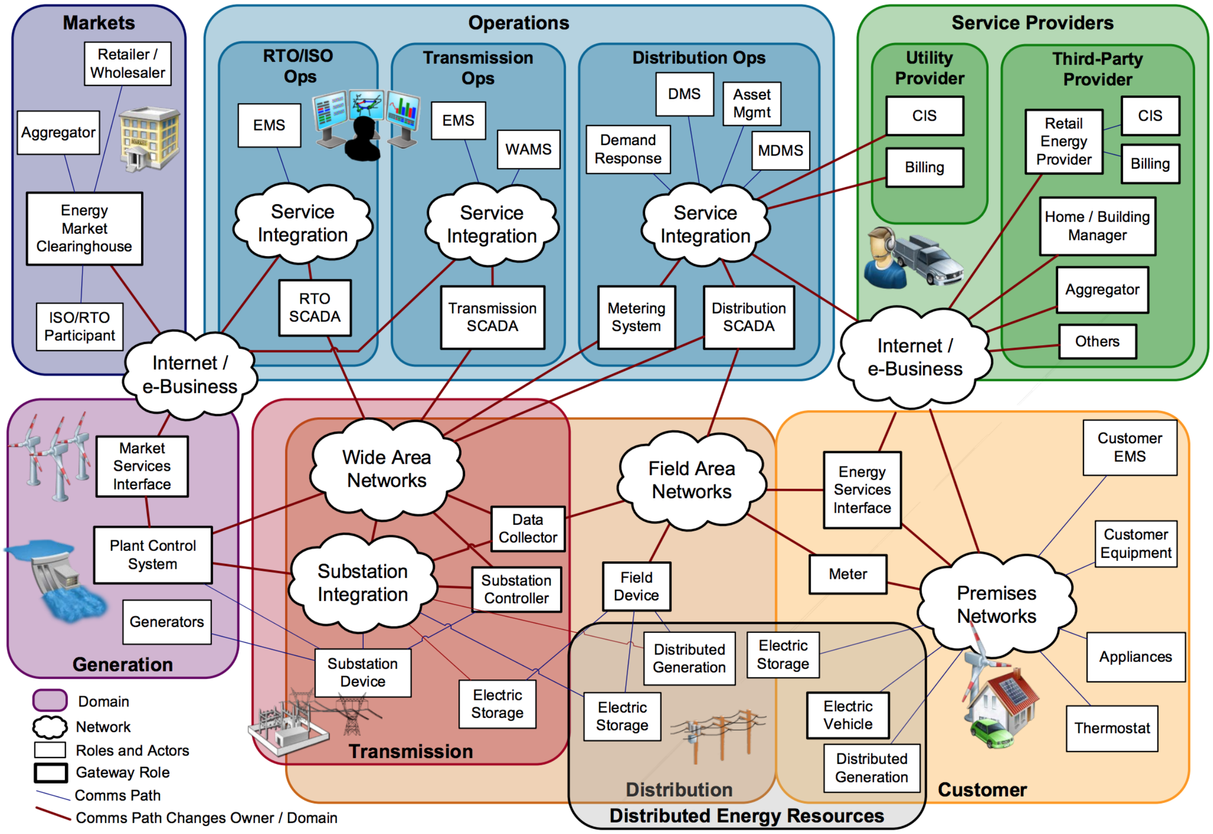


Figure 1.1: A conceptual diagram of the smart grid.

communications to deliver electricity between suppliers and consumers. As a result, smart grid introduced a number of new network infrastructures including: Wide Area Networks (WAN) that connect different entities, such as control center, generators, and markets; Field Area Networks (FAN) that connect devices within site; Premises Networks (PN) that connect customer and utility premises; and AMI Networks (or utility networks) that focus on connecting customers [50]. The new communication infrastructures should able to handle a massive amount of data associated with the smart grid applications while not interrupting functions of legacy power grids.

However, with the concern of these new developments, other vital researches must also be accomplished. Primarily, we need to devise effective technologies for protecting the availability and reliability of the envisioned smart grid communication networks and for preserving the privacy of the data that are used for the smart grid. In this dissertation, we focus on how to ensure the confidentiality, integrity, and availability of data delivery through vulnerability assessment and privacy-preserving computations in smart grid.

In smart grid, data are aggregated from diversity sources and used for different purposes such as monitoring and automatic control. One key challenge of the smart grid is how to guarantee efficiency and cyber resilience in large-scale data transmission. On the one hand, there has been an immense advancement in computing and networking technologies. Existing technologies, such as cloud computing and sensor networks, could be adapted toward the new interconnected smart grid architecture. On the other hand, those smart grid adoptions need to be carefully designed in order to provide the low latency, self-healing, reliability, and self-configurable requirement. For example, the synchrophasor network is viewed as a key enabler for power grid real-time situational awareness and control which consists of two major component: Phasor Measurement Units (PMU) is used to obtain phasor information (both magnitude and phase angle) in real time; Phasor Data Concentrator (PDC) can exchange synchrophasor data with PDCs from other locations. However, due to the low-latency demand, this network cannot directly employ other well-established network technologies. In the past decade, many researches and standards have been published for synchrophasor network. Nevertheless, fewer works have been conducted on security and privacy issues for synchrophasor networks.

Furthermore, new cybersecurity threats are particularly harmful to the power industry. Many power grid systems are less prepared for defending against cyber attacks. Specifically, many smart grid substations' components do not have an authentication mechanism of other components with which they interact; data integrity is not checked correctly; all the data that sent to the network is in plaintext. Therefore, according to [13, 82, 112, 115], smart grid communication networks are vulnerable to a variety of cyber attacks, such as Denial of Service, Man-in-the-Middle attacks. On Dec. 2015, the attackers penetrated some Ukraine electricity distribution control centers which reported to have utilized software vulnerabilities, stolen credentials, and sophisticated malware [70]. The attackers then opened a few circuit breakers and shut off power for several hours which affect more than 200,000 people. This incident led to a tremendous amount of efficiency and finance loss regarding the affected entities. To this end, it is necessary to examine the cybersecurity and privacy vulnerabilities associated with the smart grid communications.

Typically, a cyber attack over the smart grid consists of 3 stages:

- **Network Breach:** In this step, the attackers try to grant access to the power system communication networks by hacking into prosaic targets: employee's email accounts, web servers, etc. In general, the attacker can attack either one or multiple substation or control center.

- **Reconnaissance:** Before start attacking, an experienced attacker needs to fully exploit the grid system. For instance, an attacker who wants to initiate target data injection attacks needs to know the bad data detector before injecting bad data.

- **Coordinated Attack:** The attackers can attack the smart grid network by forging network packages and directly send commands to substations or bad data to control center.

In this work, we restrict our scope to the reconnaissance and coordinated attack stage. More specifically, we assume the attacker is capable of penetrating the communication network and access network traffic. For the smart grid, this assumption is reasonable because it's impossible to prevent penetration from both outside and inside coordinate. Based on

Verizon's 2018 data breach investigations report [3], 73% of cyber attacks were perpetrated by outsiders and 28% of attacks involved insiders. Some guideline for how to penetrate a network can be found in [1].

1.3 Challenges

Modern cyber-physical systems, such as the smart grid, have vulnerabilities. Although some challenges are identical to those of traditional networks with more complicated designs, various new challenges will arise with the integration of new technologies. We consider three areas in this chapter: Confidentiality, Integrity, and Availability

In the reconnaissance stage, after breaching into the internal network, a strong attacker aims to penetrate the control center or gain operational access. Compromising data confidentiality is the primary goal for the attackers in this stage. In the past, the operational systems of power industries are strictly disconnection from other networks, which causes an incredible amount of effort for cyber attackers to break in. However, this regulation becomes flexible due to the deployment of new technologies. Internal systems may interact with the public networks through multiple sources including customers to third-party providers; generation to grid operators; markets to grid operators; and third-party providers to utilities. More often, a weak attacker is only capable of controlling a limited set of internal components and communication interfaces. To maximize their interest, the attacker needs to exploit some key information of power grid; such information is either power system related (e.g., bus connections and voltage limitation) or networking related (such as IP addresses and component's identity number). Since information is hard to obtain, some attacks require long dwell time in the environment to gain sophisticated domain knowledge. Based on CrowdStrike annual cyber intrusion services casebook [31], the average attacker dwell time is 86 days. In Chapter 3, we will show that without network information, the attacker can only perform random data injection attack which can be easily detected by static false data injection detector. As such, ensuring data confidentiality is equally important for defending network breach. In this dissertation, we concern a problem that sensitive data may be disclosed to cyber attackers, in the situations where the data are processed in either private or public space.

Most of the research on the data confidentiality in the smart grid concentrate on the consumers' data privacy. Giaconi et al.[46] studied information leakage in a smart meter system, and the privacy can be partially preserved by a low-complexity policy which can approach the theoretical lower bound. Engel et al. [37] use the wavelet transform to generate a cascade of different resolutions which enable users to grant or deny access to external parties efficiently. Similar researches can be found in the literature [11, 49]. While apparently an important topic, we expand the scope of smart grid's data confidentiality research. Specifically, we investigate data confidentiality issue for operational data in the smart grid. Moreover, we are interested in protecting data confidentiality when interaction with external networks is necessary.

Data integrity and availability are the major concerns in the coordinated attack stage. In this stage, we are dealing with a situation that the adversary who has network access also obtained the necessary information. To ensure the secure interoperability across different domains and components, data integrity guarantee is the major concerns in this dissertation. Integrity involves making sure that data must be intact in transmission and further steps should be taken to ensure that data cannot be modified by any unauthorized parties (e.g., under the situation of confidentiality breach). For many applications in the smart grid, especially power system control algorithms which use data to make decisions, fabricated data can lead to misinformed controls and damaging consequences. For instance, as mentioned in [75, 100], an attacker can take advantage of the configuration of a power system to successfully bypass bad measurement detection in state estimation and mislead the operation. Availability is another critical aspect for protecting ordinary operations of smart grid which guarantees reliable service and data availability to the authorized people. To compromise availability, Denial-of-service (DoS) and distributed denial-of-service (DDoS) attacks aim to obstruct or corrupt the availability of the system by blocking data delivery between entities in the smart grid. Defending DoS attack are widely investigated in the literature [81, 104, 129]. In this dissertation, we extend these works to synchrophasor networks.

Existing information security works either provide potential vulnerabilities in communication protocol and implementation of the smart grid or high-level overviews of cybersecurity threats which focus on analyzing the impacts of cyber attacks. Most of these works are from the power system's point of views with the presence of useful attack techniques. To overcome such limitation, we provide a few real-world attacking strategies and demonstrated on physical hardware testbed in this dissertation. With the comprehensive attacking procedures, we are able to prepare better security practice.

1.4 Dissertation Outline

Following the attacking scenario for the smart grid, the contributions of this dissertation are summarized as follows:

- We investigate the vulnerability of current synchrophasor networks and provides mitigation analysis (protecting data availability and integrity)
- A novel method to detect False Data Injection Attack using deep learning is proposed (protecting data integrity)
- A privacy-preserving algorithm for large scale security constrained optimal power flow is presented (protecting data confidentiality)
- Dynamic simulations leveraging the cloud computing is presented in the privacy-preserving form (protecting data confidentiality)

Chapter 2 starts with investigating the vulnerability of current synchrophasor networks. With the help of synchrophasor networks (or PMU networks), the system dynamic is achieving situational awareness through a wide area measurement system which plays an indispensable role for the future smart grid. To build a stable and cyber resilience synchrophasor network, it is necessary to examine the cyber-physical security of current synchrophasor networks. Comprehensive security study is revealed for conventional information communication networks [110] and a few works have been conducted for exploring the vulnerability of synchrophasor networks [115, 13, 82, 112]. Nevertheless, there lacks practical attacking schemes and mitigation analysis for synchrophasor networks in the literature. In Chapter 2, we first summarized the existing vulnerabilities of synchrophasor networks. Based on the found security vulnerabilities, we proposed 3 attacking schemes over synchrophasor network located in CURENT [1] hardware testbed which include sniffing, false data injection, and Denial-of-Service (DoS) attacks. For each possible attack, we discuss practical defense mechanisms in the real-world settings.

Followed by cyber-physical security analysis, Chapter 3 introduces a novel method to detect false data injection attack using deep learning. While all the examined cyber attacks are of great importance, we found that false data injection attack can bypass bad data detection and stealthily insert any bias value into the estimated state which is particularly harmful to the smart grid. For instance, false data injection attack can coordinate with physical attacks to cover line outages initiated by physical attacks [69]. False data injection attack is first introduced by Liu et al. [75] targeting the state estimation in the power grid. One way of detecting false data injection attacks is to secure basic measurements which are selected strategically. However, based on our vulnerabilities assessment, it is not suggested to trust some measurements completely. Moreover, PMUs are able to provide synchronous measurement data which motivates us to develop an anomaly detection system based on the neural network to enable the construction of a smart grid specific defense mechanism. In our proposed solution, we make use of both PMU measurement and network traffic. By doing so, our model learns normal behavior from normal data and is unrelated to certain attacks, and thus capable of detecting unseen attacks. We fully implement the proposed scheme over IEEE 39-bus system, and it is shown that our scheme can identify anomalies which cannot be detected by traditional static bad data detection.

We start the discussion of protecting data confidentiality in Chapter 4. Unlike previous works for protecting consumer data privacy that raised great concern recently, this dissertation mainly focuses on preserving power operation data confidentiality. In our cyber-physical security and false data injection attack researches, we found that a successful cyber attack often requires sensitive power grid data which can be captured by cyber attackers

[1] CURENT, Center for Ultra-Wide-Area Resilient Electric Energy Transmission Networks, is a National Science Foundation Engineering Research Center that is jointly supported by NSF (National Science Foundation) and the DoE (Department of Energy).

or leaked internally. While traditional cryptography is useful to scramble a plaintext in such a way that any interceptor of this ciphertext cannot know the real value, the cipher still needs to be decrypted before the computation phase. In this way, the security of data confidentiality can be dramatically increased if sensitive data remained encrypted during the computation phase. Chapter 4 starts with preserving data confidentiality for large-scale security constrained optimal power flow (SCOPF). In actual operations, optimal power flow of the entire distribution network often requires to be solved in real time (every five minutes for many Independent System Operators) to ensure demand is met accurately. This provides great opportunities for attackers as optimal power flow usually involves multiple entities, and the computations require data from a variety of trust domains. To mitigate this threat, each substation encrypts their private data after which a third party performs our privacy-preserving SCOPF algorithm over the encrypted data without decrypting it. To facilitate the computation process, we decompose the SCOPF problem into independent subproblems using ADMM and gradient projection. The proposed privacy-preserving algorithm is also tested with both IEEE 57-bus system and IEEE 118-bus system. Security analysis and performance evaluations are also carried out.

Next in Chapter 5, we describe another scenario that an entity wishes to outsource some heavy computations to the cloud (either a private cloud or public cloud) without leak any data to the cloud. This scenario is motivated by previous works [59, 117] that showed the possibility to conduct power system simulations by efficiently outsourcing the computations to the public cloud. Yue Tong et al. [117] alleviate the privacy concern by a solution of outsourcing of power system dynamic simulations with disguising and code obfuscation technologies. However, both disguising and code obfuscation do not have strong security guarantee. For example, a chosen-ciphertext attack can be used to recover injective mappings. We future enhance the security of dynamic simulation by proposing a privacy-preserving dynamic simulation using homomorphic encryption. By doing so, both the input sensitive data and the final simulation result are encrypted while the majority computation is protected by our semantic secure algorithm. This algorithm provides confidence for utility companies or ISO to off-load dynamic simulations to the public domains. The proposed scheme was implemented and evaluated with the IEEE 3-Machine 9-Bus and analysis of the trade-off between feasibility, efficiency, and security for homomorphic encryption and trajectory disguising is discussed afterward.

Finally, Chapter 6 summarized the work in this dissertation and suggestions for future work are provided.

Chapter 2

Cyber-physical Security of the Synchrophasor Networks

2.1 Introduction

A smart grid revitalizes the legacy power grid with the benefits of modern communications to achieve real-time monitoring and enable the fast balance of supply-demand management. Naively, smart grid devices can be directly connected to the Internet. However, as transmission latency, reliability and bandwidth requirements for different components vary widely, a number of communication systems and network structures have been proposed for the smart grid. For example, advanced metering infrastructure (AMI) is a system that integrated with communications networks, data management, and smart meters to permit communications between utility companies and customers. With AMI, utilities shift from reading meters to remotely maintaining and controlling key aspects of the grid. Existing AMI communication networks are based on a hybrid of networking technologies including Ethernet, power line communication (PLC), and a variety of wireless technologies such as WLAN and LTE. Thus, the heavy reliance on data availability and integrity for smart grid poses a great challenge on respective communication networks.

In the past, legacy power system was designed for local operations with limited wide controls. The security of the legacy power grid largely relies on its physical isolation from the public networks, i.e., the airgappedness. For smart grid, most communication infrastructures are developed to support monitoring and controlling applications, e.g. Supervisory Control and Data Acquisition (SCADA), Energy Management Systems (EMS), Distribution Management Systems (DMS), etc. For SCADA system, It is designed for distribution automation and remote controls, and it can benefit power utilities to achieve higher supply reliability with low costs. Additionally, connections between distribution automation or DMS control centers and EMS are typically provided via high-capacity IP Gateways [55]. As a result, the air-gap between the power grid internal network and the Internet has been increasingly blurring [128]. Therefore, even with defending mechanisms (i.e., Intrusion Detection System (IDS) and firewall) deployed, the smart grid communication network can no longer be deemed as physical isolation. This leads to significant issues related to privacy and security on the power grid. For instance, false data may mislead operation and control functions of the control and monitor system such as EMSs. In a worst-case situation, carefully fabricated data could result in potentially catastrophic consequences as suggested in [71]. In the past decade, there are a increasing number of cyber-attacks targeting power grid:

- The communication networks of power and utility companies faced around 12 cyber attacks between 2004 and 2008 [102]. This is sharply increased by 20 %. As the SCADA becoming essential for smart grid infrastructure, the cybersecurity concerns are increased rapidly.

- In 2010, a computer worm called 'Stuxnet' was first discovered which intend to attack industrial software and equipment and spreads through Windows operating systems [41]. This type of cyber attack is based on the computer virus and can targets critical infrastructures of industrial cyber-physical systems which introduces new threads to power systems [67].

- On December 2015, a report is released to describe a power outages caused by cyber attacks. The internal networks of three energy distribution companies in Ukraine are successfully compromised by the external hackers and the electricity supply is temporarily blocked to the end consumers [70].

As the smart grid proliferating, before trusting a newly introduced network-related technology, we need to assess its potential vulnerabilities of all aspects, including the protocols, implementations, and configurations, etc. In this chapter, we chose the synchrophasor network. This is because of its imperative role for the future smart grid which offers significant advantages by providing fast and precise voltage and current phasors measurement of the entire grid. In contrast, traditional SCADA system are unable to provide a synchronized real-time assessment of the system as a result of low sampling rates and lacks of time synchronization for wide area monitoring systems (WAMS). What's more, synchrophasor networks have not been under comprehensive vulnerability assessment in the real world. Only a few works have been conducted for exploring the vulnerability of synchrophasor networks [115, 13, 82, 112] in theory or experimental setup. In this chapter, we attempt to gain more insights into how to attack a synchrophasor network starting from scratch on CURENT hardware testbed. We consider three areas in this chapter: Confidentiality, Integrity, and Availability. Based on the attempts, practical defense analysis against cyber attacks are presented.

2.1.1 Challenges

Synchrophasor networks serve as a critical role for the traditional power system to emanate as a cyber-physical system. In the meantime, as the essential energy source, our life can be significantly impacted by a successful cyber attack against the power grid. Power generation, distribution, and transmission networks should be taken into consideration in this situation. It is infeasible to examine all the necessary strategies for improving the cyber-physical security of the smart grid without those components.

However, due to security and privacy concerns, utility companies tend to trust the airgappedness and traditional security practice, thus ignore new cyber-physical security methods and assessment. As a result, there is a lack of cybersecurity research that performs on real smart grid cyber-physical system. Indeed, researchers usually leverage simulation environments and IEEE test systems to identify the attack surface, target, and attack vector. Although this type of evaluation setup is sufficient under specific circumstance, it neglects some essential factors. For instance, a fix estimated delays are specified for communication networks based on WAMS needs and specifications in [5]. However, network delays are dynamic and vary according to a number of aspects. Additionally, some research efforts have been conducted to develop cyber-physical integrated power system testbed including GE Grid Solutions Smart City Testbed (SCT) at Washington State University, PowerCyber testbed at Iowa State University, and CURENT center at University of Tennessee, Knoxville. Despite that some theoretical works [112, 51] have been carried out, it is still challenging to conduct cyber attacks from the attackers' point of view. As such, most papers only consider a simplified threat model, such that some vital information is supposed to be captured by the attacker. In contrast, we are motivated to develop cyber attacks targeting synchrophasor networks that require no prior knowledge and minimum network access. Therefore, more effective cyber-physical impacts and security analysis can be provided.

2.1.2 Outlines

This chapter is organized as follows. Section 2.2 reviews existing research concerning the cybersecurity of synchrophasor networks. We then revisit the concept, existing standard, and the cyber threat of synchrophasor network in Section 2.3. Then, cyber attacks paradigm against real-world synchrophasor network has been discussed in Section 2.4. Additionally, we present the evaluation setup and venerability assessment in Section 2.5 and test all proposed attacks in Section 2.6. Especially, evaluations have been conducted over both small prototype synchrophasor (Section 2.7.4) and real-world emulation in CURENT hardware testbed (Section 2.5.1). Finally, mitigation analysis is provided in Section 2.7.

2.2 Related Work

Cyber-physical security (CPS) is critical to address the requirements of a sophisticated smart grid. Some researchers reviewed cyber attacks, countermeasures, and challenges of smart grid cyber-physical system. C. Beasley et al. [13] provide a review of a wide selection of cyber attacks against synchrophasor networks and group them into five categories, i.e., interruption attacks, interception, modification, and fabrication. Countermeasures and future research directions are also provided for each attack. Similar work is done by [68] which studies the smart grid network architecture and introduces major security challenges. Smart grid security fundamentals are discussed to defend existing or future malicious attacks. More specifically, state-of-art role-based access control, authentication mechanism, privacy-preserving computations, and intrusion detection system are listed and evaluated to resist basic cyber attacks against smart grid architecture. Y. Mo et al. [82] first analyze both cyber and system-theoretic approaches and show that they are essential to improve the security of smart grids than traditional methods. The authors provide an example that a cyber attack on the data integrity can be mitigated by system-theoretic approaches.

In order to identify vulnerabilities in synchrophasor networks, a variety of research on vulnerability assessment is conducted to analyze the weaknesses of specific network or device. Y. Tong [115] conducts thorough penetration testings through a small scale synchrophasor network to discover network vulnerabilities and improve the security. By testing a variety of technologies including packet sniffing, packet injection, fuzzing, the author identified few vulnerabilities associated with the IEEE C37.118. T. Morris et al. [84] use an MU Dynamics MU-4000 Analyzer to perform denial-of-service, network congestion, and protocol mutation tests for synchrophasor network. The result shows that some devices may finally becomes unresponsive when receiving high volume of packets. Some devices reset themselves during the test, and become available when the packet rate returns to certain levels. C.C. Sun et al. [112] utilized Washington State University (WSU) testbed and the Smart City Testbed (SCT) to test and analyze basic defense mechanisms against cyber attacks targeting substations. A hybrid anomaly detection system is deployed to examine the anomaly detection and mitigation capabilities.

2.3 Preliminary

2.3.1 Synchrophasor

A phasor is a sinusoidal signal $\hat{A} = Acos(\omega x + \phi) = (A, \omega, \phi)$ can be represented by a cosine function with a magnitude A, frequency ω, and phase ϕ. A synchrophasor is a phasor measurement corresponding to a synchronized time. With synchrophasor measurement, we can determine the absolute phase-angles between phase quantities at different locations on the power system. High resolution synchronized data is critical for the smart grid. By synchronizing the sampling processes for different signals which may be captured hundreds of miles away, it is possible for monitoring the dynamics of the power grid. For instance, it is feasible to track the system dynamic response to generator tripping events with synchrophasors [36].

2.3.2 Phasor Measurement Unit and Phasor Data Concentrator

Synchrophasors are synchronized measurements that contain both the magnitude and phase angle of the signals in power system. The synchronization is attained by a real-time sampling utilizing calibrating signals from the Global Positioning System (GPS) technology. Synchrophasors are measured by high-speed monitors called Phasor Measurement Units (PMUs) which are estimated 100 times faster than SCADA. A comparison between SCADA and PMU is listed in Table 2.1. PMU measurements report high accuracy grid states and offer strong insight into grid stability. A typical PMU measures:

- Positive sequence voltages and currents
- Phase voltages and currents
- Local frequency
- Local rate of change of frequency
- Circuit breaker and switch status

With all the measurements, system operators are capable to monitor the power system across a wide area and estimate grid states in real time, thus to identify and react to emerging events. Phasor data concentrator (PDC) is utilized to align data from multiple PMU devices by the synchronized time and sends aggregated synchrophasor data as a single data frame. Several ongoing projects such as the North America Synchrophasor Initiative (NASPI) are dealing with the research and development of synchrophasor networks [32].

[1] Based on SEL latest production.

Table 2.1: Comparison between SCADA and PMU

	SCADA	**PMU**
Sample Rate	1 sample every 2-4 Seconds	30 samples per second (up to 60 samples per second)
Measurement	Magnitude	Magnitude and phase angle
Synchronization	No	Using GPS
Input/output Channels	100+ Analog & Digital	Up to 64 Phasors, 30 digital, 64 analog [1]
Observability	Steady State	Dynamic states
Monitoring	Local	Wide Area

Figure 2.1 illustrates a typical architecture of the synchrophasor network. PMUs are placed at desirable locations by utility companies to achieve best state estimation performance. PDC receives data measurement from PMUs within the same utility where quick decisions are made (less than 100ms). The PDC may forward the aggregated data to a central PDC called SuperPDC, usually hosted by the regional Independent System Operator (ISO), for higher-level control (100ms-1s decisions). According to IEEE Standard for Synchrophasor Data[57], additional functions may be provided as follows:

- Various quality inspections on the phasor data and insertion of appropriate flags into the correlated data stream.
- Checking for disturbance flags and recordings of data files for analysis.
- Monitoring of the overall measurement system and displaying the results, as well as the recording of the performance.
- Number of specific outputs, such as a direct interface to a SCADA or EMS system.

2.3.3 Synchrophasor Network Standards

For the synchrophasor network, a standard defines an open design for all vendors to make use of the synchrophasor more efficient and reliable, which can facilitate data exchange between devices, data collection, and various applications. It can be used directly or align

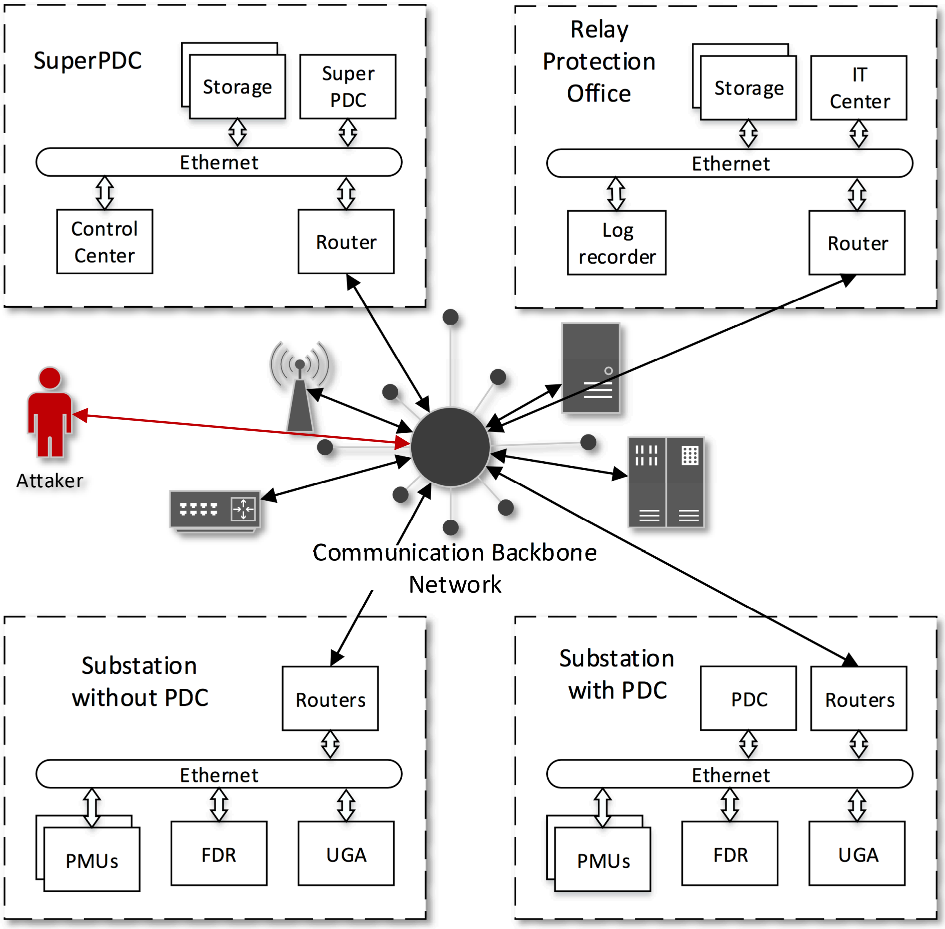


Figure 2.1: An example of architecture of a synchrophasor network.

with other communication protocols which is the most direct method for data transmission and accumulation for wide area monitor system. Few standards have been proposed for the data transmission between PMUs and PDCs. The IEEE 1344 standard for synchrophasors was first published in 1995 and revised in 2001. It was replaced by IEEE C37.118.2-2011 [58] in 2011 to deal with issues concerning the use of PMUs which is a complete revision of the previous standard. In 2012, a new part of IEC 61850, IEC 61850-90-5 [56], specifying the IEEE C37.118 based synchrophasors' protocol with respect to IEC 61850, is proposed for PMU communication networks. Although IEC 61850-90-5 has several unique features over C37.118 [61], however, its adoption is still inadequate and thorough investigations of its features and requirements are limited. In this dissertation, we still use C37.118 and IP over Ethernet as the main communication protocol for synchrophasor network.

Based on [57], IEEE C37.118 is split into two standards: one concerning measurement requirements and the another concerning the data transfer requirements. These two standards are used to improve synchrophasor measurement technology in order to simplify integration with other communication protocols for synchrophasor measurements.

A typical communication scenario between to PMUs and a local PDC is depicted in Figure 2.2. For simplicity, we do not consider the header message which may be requested by the control center using command messages. Based on [57], IEEE C37.118 standard defines four types of messages: data, configuration, header, and command, which are briefly introduced as follows:

- **Data Frame** transfers real-time measurements data from PMU or PDC to the receiving device (PDC or SuperPDC, correspondingly).

- **Configuration Frame** is sent by PMU/PDC to notify the receiving device the configuration information of the data message, including the number of channels, types, and scaling factor, etc.. Configuration frames are intended to be read by machines.

- **Command Frame** is sent from a data concentrator (a PDC or a SuperPDC) to its source devices (PMUs or PDCs, correspondingly) to coordinate the communication (start/stop data transfer, request for configuration frames, etc.).

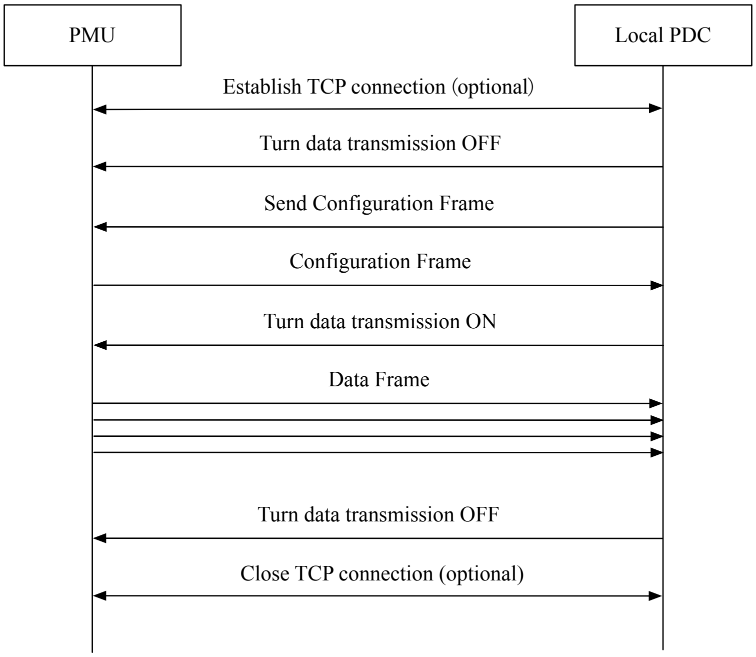

PMU
Local PDC
Establish TCP connection (optional)
Turn data transmission OFF
Send Configuration Frame
Configuration Frame
Turn data transmission ON
Data Frame
Turn data transmission OFF
Close TCP connection (optional)

Figure 2.2: A typical IEEE C37.118 communication scenario for data transmission operating in commanded mode.

- **Header** information is human readable descriptive information sent from the PMU/PDC which is provided by the user.

2.4 Cyber Attacks Targeting Synchrophasor Networks

As discussed earlier, IEEE C37.118 usually uses TCP/UDP, IP, over Ethernet as the low level communication technology. As a result, any network attacks targeting these underlying technologies are also possible on the synchrophasor network. However, since related attacks and respective countermeasures have been well studied in the network security research community [23, 79, 109], and that there even exist automated tools are available for testing relevant vulnerabilities, we put our focus on the security of the application layer and above, i.e. the IEEE C37.118 protocol and implementations of PMUs and PDCs.

To better understand feasible attacks targeting the synchrophasor network. We first introduce the CIA triad which is a model that helps us implement information security programs to protect their confidential data. The CIA triad comprises three parts:

- **Confidentiality:** Information should only be restricted to entities who have legal access to it.
- **Integrity:** Data stream must be protected from unauthorized modification and destruction.
- **Availability:** Availability guarantee that authorized persons can access to the data when necessary.

2.4.1 Compromising Data Availability

Data availability of the synchrophasor network is primarily subject to denial-of-service (DoS) attacks if the supporting communication facilities fail to identify unexpected frames. By saturating the victims' communication resources or by obstructing the communication channel between the legitimate client and server (e.g., jamming the wireless channel) [60], the attackers can slow down or block the normal data transmission. In general, DoS attacks fall into three basic categories based on the attacking volume and the targeting vulnerabilities: volumetric attacks, which utilize high traffic to flood the network bandwidth; protocol attacks, which focus on exploiting weakness in layer 3 and 4 ; application attacks, which target on individual applications and are considered the most challenging attacks to be mitigated.

In synchrophasor network, DOS attacks can target any relevant protocols, range from the network layer (ARP, ICMP, IP), transport layer (TCP, UDP), or the application layer (IEEE C37.118). Moreover, the target of DoS attack can be the communication between PMUs and local PDC, local PDCs and superPDC, or superPDC and the control center. The most critical choice for DoS attacks is targeting communication channel between superPDC and the control center. This attack will result in a complete loss of visibility of substations for the control center. The impact of compromised data availability can also vary according to types of frames loss. Loss of command frame will prevent the control centers from controlling various devices. Loss of configuration frame will make PDC unable to decode upcoming data frames. Loss of data frames will leave the grid operators with inadequate information about the system dynamics, thus, obstructing the operators to make decisions.

In this research effort, because DoS attacks can be mounted by exploiting various vulnerabilities, we only focus on testing DoS attack on transport layer over communications between substation PDC and PMUs. Besides compromising data availability, we show that DoS attacks can be leveraged for other attacks including compromising data confidentiality.

2.4.2 Compromising Data Confidentiality

The pivot concern over data privacy is at the consumer end. As the smart grid connect end customers with utilities and as customers increasingly participate in managing their energy, data confidentiality and privacy has become a critical concern. For the synchrophasor network, data confidentiality is also a big issue as most high-level cyber attacks require domain knowledge. Through eavesdropping the network traffic of the synchrophasor network, attackers can learn substation name and ID, components' (e.g., PMU, breakers) location and measurements, and configurations of the individual devices. Since C37.118 does not specify encryption scheme, all C37.118 packets transmitted through synchrophasor network are carried in plaintext. The confidentiality concerns with respect to the smart grid is summarized as below:

- Perform real-time surveillance. The utilities collect data measurements for high-level monitoring and other services development. The data interception can be considered as the real-time surveillance if the potential adversaries can capture short-interval data streams.

- Power system operating data have high economic value. An attacker can sell that information to competitors or the black market to gain economy interest.

- Sensitive data can help the attacker for a more sophistic attack. For example, with measurement Jacobian matrix, a false data injection attacker can pass the commonly residue-based bad data detection.
- A data breach can have serious regulatory and reputational impact, but disruption to the BES could arguably cause far greater harm to business performance, national security, and public safety.

The U.S. government created Protected Critical Infrastructure Information (PCII) Program to regularize the sharing of critical information between private sector infrastructure and operators, such as utilities and the administrations, to prevent the data breach. However, such a program cannot theoretically stop attackers from compromising data confidentiality. In Section 2.7, we propose a lightweight encryption protocol based on C37.118 and test its overhead for synchrophasor network.

2.4.3 Compromising Data Integrity

The integrity attacks can target either customer's information (e.g., customer account balance, power pricing) or network operation information (e.g., synchrophasor measurement, command frames). Particularly, such attacks can deliberately modify the original measurement in the smart grid in order to mislead some critical control algorithms. The impact of attacks targeting data integrity in the power grid is vital. A notable work is by Y. Liu et al. [75], which proposed a type of false data injection attacks against the state estimation in the power grid.

To launch a false data attack in the synchrophasor network (detailed in Section 2.6.3), the attacker first captures a legitimate IEEE C37.118 data frame sent from a source device by packet sniffing. With the obtained information, the attacker can construct a fake message containing false measurement data to imitate an authentic message. The attacker then injects the fake message, which will be routed to the PDC. Consequently, the PDC will accept the fabricated measurements if there is no message or user authentication mechanism or such mechanism is weak or not in use. If the false data is deliberately chosen, it may mislead power grid controls and decision-making that uses the synchrophasor data as the input. Some works [125, 34, 71] demonstrate how deliberately chosen false data can mislead the state estimation and cause disastrous consequence, which, however, does not specify how to inject false data into the synchrophasor network. In Section 2.6.3, a practical false data injection attack scheme is introduced which shows the impact of compromising data integrity for the smart grid.

Current mitigation methods over false data attack are protecting set of basic measurements [7], PMU-based protection [29], and proposing new detection algorithms [132]. We will present a new way of detecting false data injection attack based on deep learning in Chapter 3.

2.5 Vulnerability Assessment for Synchrophasor Networks

2.5.1 Evaluation Setup

To validate our proposed system in the real-world scenario, we explore the vulnerability on hardware testbed located in the CURENT center as shown in Figure 2.3. As an alternative to the actual power grid cyber-physical system, a testbed can be used to examine the implementation and impact of cyber attacks. CURENT hardware testbed provides broad time scales in one system - microseconds for power electronics devices and milliseconds to seconds for power system event. It also integrates real-time monitoring, protection, and control.

As shown in Figure 2.4, the hardware testbed is based on an aggregated WECC system. A remote load center including L12 and L13 is fed by a local generator, two inter-connected systems, and offshore wind through multi-terminal HVDC (MTDC). Four PMUs are placed at each load bus, and synchrophasors measurements are collected by local PDC. PDC then aggregates synchrophasor measurements and sends it to the control for wide-area monitoring and control. CURENT hardware testbed uses latest version openPDC to emulate local PDC and LabVIEW to mimic the control centers in power systems. For conducting the vulnerability assessment, a machine running Kali Linux OS is deployed to emulate the cyber

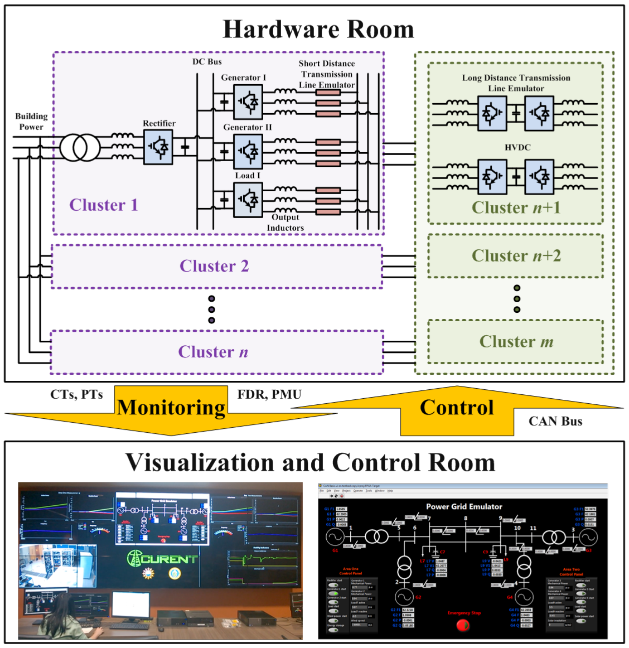


Figure 2.3: Power electronic converter based Hardware Universal Grid Emulator setup in CURENT center.

attacker who has network access. The packet manipulation program Scapy 2.2.0 is used in this research for generating attacking packets. Wireshark is used to analyze and visualize network traffics.

2.5.2 Vulnerability Exploration

The goal of the exploration phase is to validate the possible weaknesses of the PMU network, which are deduced from the information leaked from the reconnaissance phase. Evaluating the information gathered in the reconnaissance phase and also the public information about the C37.118 standards, we list vulnerabilities that we will manually explore and validate during the exploration phase in Table 2.2.

We briefly explain the listed vulnerabilities as follows. 1) as we have already discovered in the reconnaissance phase, all C37.118 frames are transferred in clear. Hence, not only can attackers intercept and eavesdrop configuration frames but it is also possible for attackers to monitor for command and data frames. 2) as the C37.118 standard does not specify any user authentication mechanism, it is possible for the attackers to impersonate a legitimate publishing or subscribing devices and confuse, mislead, or sabotage other parties in the PMU

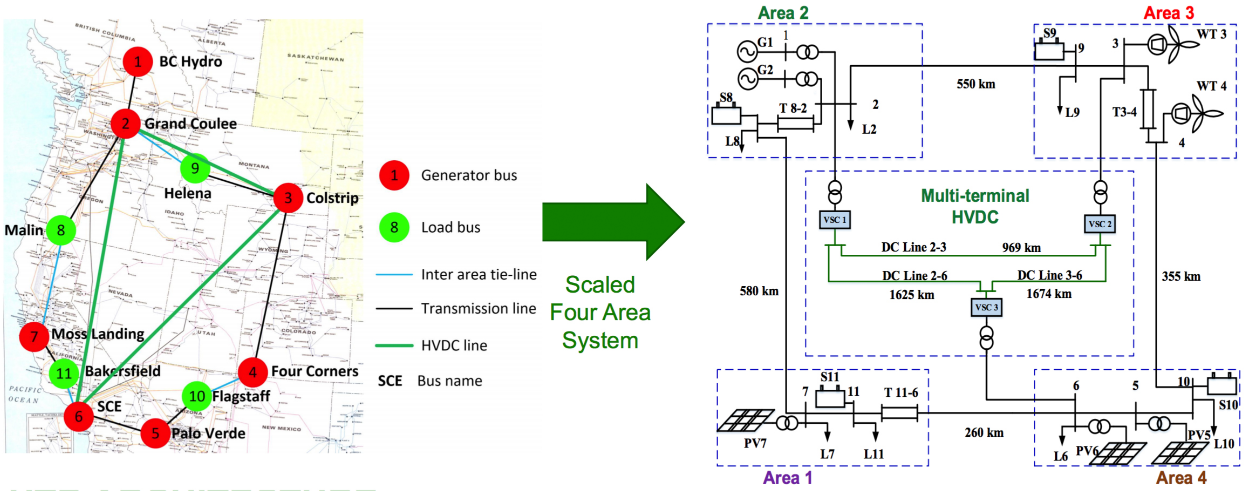


Figure 2.4: WECC system in CURENT hardware testbed.

Table 2.2: Vulnerabilities of CURENT hardware testbed

Cause of Vulnerabilities	**Possible Attacks**	**Testing Technique**
Lack of encryption	Eavesdropping, Replay	Packet sniffing
Lack of user authentication	Impersonation man-in-the-middle attacks	Packet sniffing Packet injection
Lack of message authentication	False data injection attacks	Packet sniffing Packet injection
Unexpected frames	Denial-of-Service	Fuzzing

network. 3) as the case of lacking user authentication, neither C37.118 includes any message authentication mechanism. As a consequence, all frames are subject to frame modifications; a receiving device is unable to distinguish legitimated frames and modified frames. We use packet sniffing and packet injection to validate this vulnerability. 4) as a stateful protocol, a device that runs C37.118 protocol manages its transition of states based on its current state and the frames it receives. If the incoming frames are expected under the current state, the device should make the state transitions accordingly. If not, the device should also handle for the case properly.

2.6 Exploit Development for Synchrophasor Networks

2.6.1 Denial-of-Service Attacks

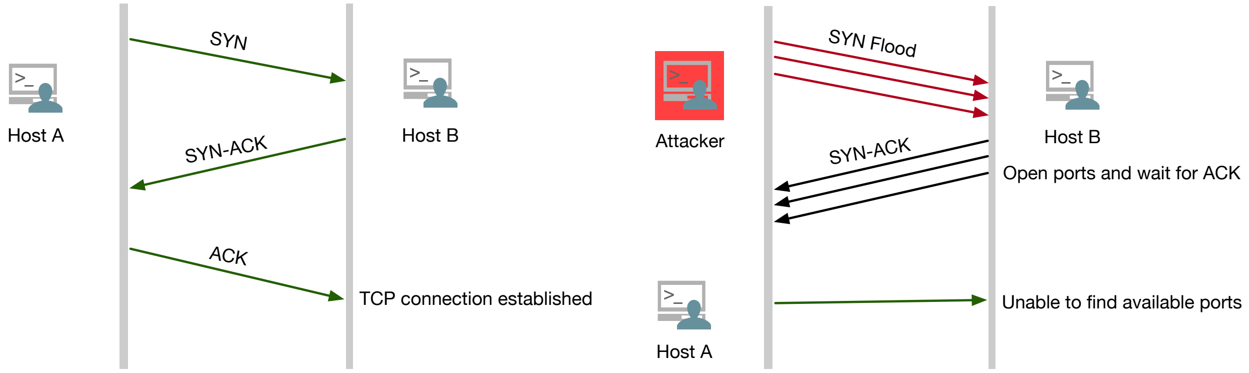
Generally, there are two types of DoS attacks: flooding or crashing services. Crashing services directly exploit the weakness that can make the system or service to fail. In such attacks, the messages are sent to the victim to takes advantage of bugs in software or hardware so that it can't be reached or utilized by the legitimate users.

Alternatively, flood attacks initiate by sending large traffic to the server, causing them to slow down or unresponsive. Popular flood attacks include:

- Buffer overflow attacks: The idea is to send more packages to a network destination than the system initial configured to handle. It differ from other DoS attacks that are designed to exploit bugs specific to particular applications or protocols and is considered to be the most common DoS attack.

- Ping flood, also known as ICMP flood: It leverages misconfigured network devices by overwhelming a targeted device with ICMP echo-request packets. Both outgoing bandwidth and incoming bandwidth are consumed in this case.

- SYN flood: An attacker sends an SYN request to a server, but never reply ACK. It continues until victim's ports are saturated with malicious requests, and unavailable for authentic users to connect.


Host A
SYN
Host B
SYN-ACK
ACK
TCP connection established
Attacker
SYN Flood
Host B
SYN-ACK
Open ports and wait for ACK
Host A
Unable to find available ports

(a) Normal three-way handshake **(b)** SYN flood attack

Figure 2.5: The progression of a TCP three-way handshake compared with a SYN flood attack.

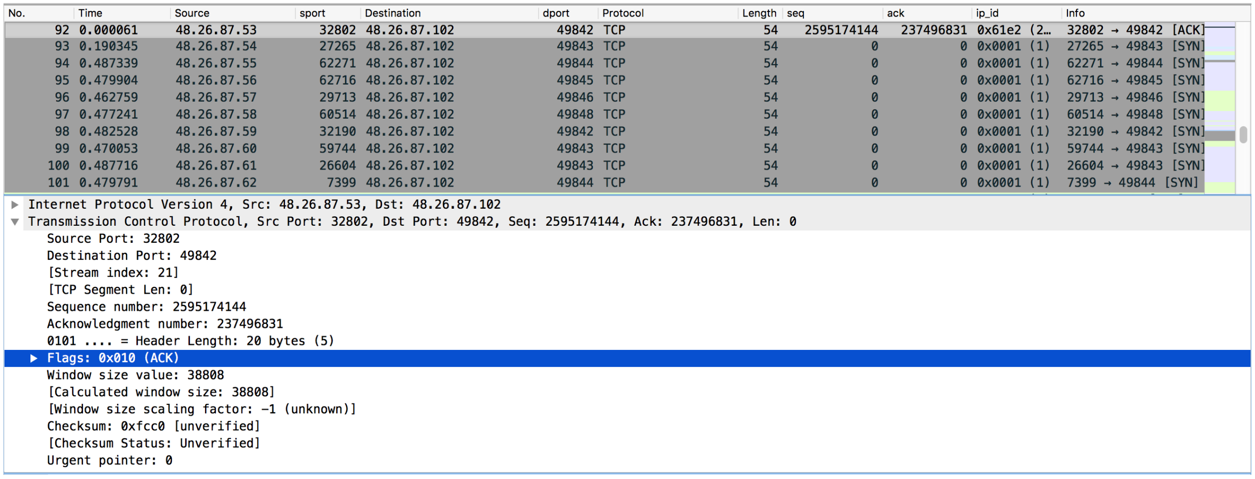
T. Morris et al. [84] tested various DoS attacks such as transport layer attacks, ICMP attacks against a PMU network. It shows that all devices tested eventually became unresponsive when the traffic volume increases beyond that devices ability to process packets. In this work, we do not restrict how to implement various DoS attacks. Instead, we focus on the impact of DoS attacks for synchrophasor networks and only test transport layer attacks as our main attacking method.

SYN Flood Attacks

When two hosts establish a normal TCP “three-way handshake” which follows three steps (Figure 2.5a):

1. Host A requests connection by sending SYN (synchronize) message to Host B.
2. Host B replies with SYN-ACK (synchronize-acknowledge) message to Host A.
3. Host A finally responds with an ACK (acknowledge) message, and a TCP connection is established.

In an SYN flood attack which is shown in Figure 2.5b, the attacker sends flooding SYN packets to all TCP port on the victim with fabricated source IP address. The victim receives multiple SYN requests to establish TCP connections and responds to each request with an SYN-ACK packet. The victim will wait for acknowledgment of its SYN-ACK packet for a



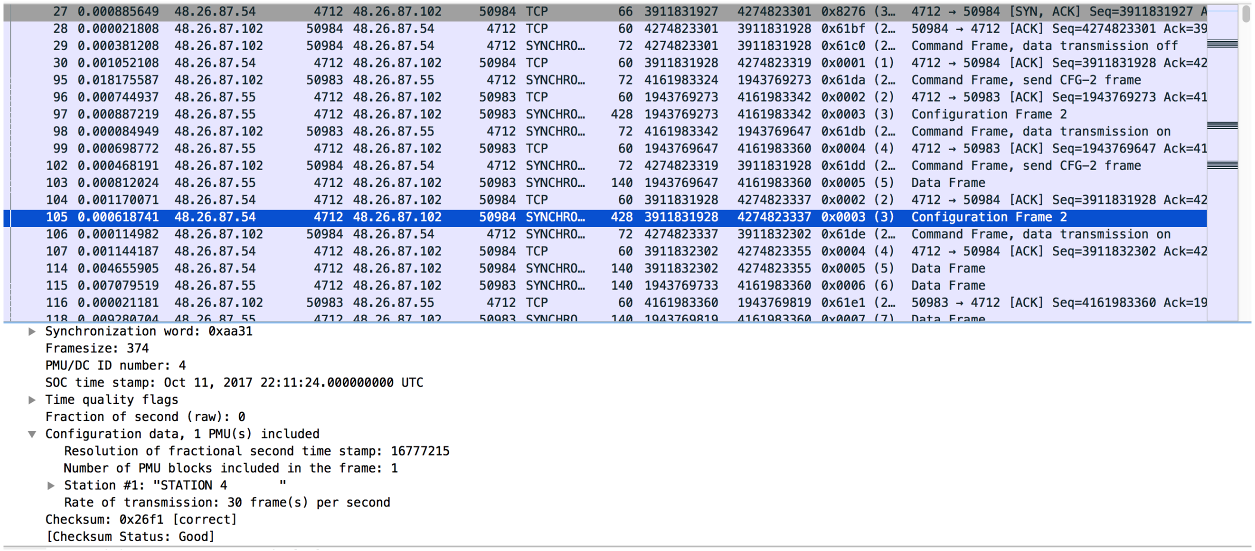
```
No.   Time      Source        sport  Destination   dport  Protocol  Length  seq         ack        ip_id         Info
   92 0.000061  48.26.87.53   32802  48.26.87.102  49842  TCP           54  2595174144  237496831  0x61e2 (2…    32802 → 49842 [ACK]
   93 0.190345  48.26.87.54   27265  48.26.87.102  49843  TCP           54           0          0  0x0001 (1)    27265 → 49843 [SYN]
   94 0.487339  48.26.87.55   62271  48.26.87.102  49844  TCP           54           0          0  0x0001 (1)    62271 → 49844 [SYN]
   95 0.479904  48.26.87.56   62716  48.26.87.102  49845  TCP           54           0          0  0x0001 (1)    62716 → 49845 [SYN]
   96 0.462759  48.26.87.57   29713  48.26.87.102  49846  TCP           54           0          0  0x0001 (1)    29713 → 49846 [SYN]
   97 0.477241  48.26.87.58   60514  48.26.87.102  49848  TCP           54           0          0  0x0001 (1)    60514 → 49848 [SYN]
   98 0.482528  48.26.87.59   32190  48.26.87.102  49842  TCP           54           0          0  0x0001 (1)    32190 → 49842 [SYN]
   99 0.470053  48.26.87.60   59744  48.26.87.102  49843  TCP           54           0          0  0x0001 (1)    59744 → 49843 [SYN]
  100 0.487716  48.26.87.61   26604  48.26.87.102  49843  TCP           54           0          0  0x0001 (1)    26604 → 49843 [SYN]
  101 0.479791  48.26.87.62    7399  48.26.87.102  49844  TCP           54           0          0  0x0001 (1)    7399 → 49844 [SYN]

▶ Internet Protocol Version 4, Src: 48.26.87.53, Dst: 48.26.87.102
▼ Transmission Control Protocol, Src Port: 32802, Dst Port: 49842, Seq: 2595174144, Ack: 237496831, Len: 0
    Source Port: 32802
    Destination Port: 49842
    [Stream index: 21]
    [TCP Segment Len: 0]
    Sequence number: 2595174144
    Acknowledgment number: 237496831
    0101 .... = Header Length: 20 bytes (5)
  ▶ Flags: 0x010 (ACK)
    Window size value: 38808
    [Calculated window size: 38808]
    [Window size scaling factor: -1 (unknown)]
    Checksum: 0xfcc0 [unverified]
    [Checksum Status: Unverified]
    Urgent pointer: 0
```

Figure 2.6: SYN flood attack targeting CURENT hardware testbed.

preset of time. During this time, the open ports stay open and the victim cannot close them with RST packets. Before the connection times out, if another SYN packet arrives, it will leave some connections half-open. This is the reason why SYN flood attacks are also referred to as "half-open" attacks.

Attacking Local PDC

We demonstrate SYN flood attacks on CURENT hardware testbed. Figure 2.6 shows how does an attacker generate attacking packets. Here, 48.26.87.102 is the IP address of PDC and port 49842 to 49848 is the targeting ports for the attack. We notice that, after the attacks, the attacker is able to slow down and obstruct the communications between PMUs and PDC. Due to limited attacking power, DoS attacks cannot completely block the data transmission. In fact, SYN flood attacks are often initiated by the Distributed denial-of-service (DDoS) attackers which overwhelm the target with traffic from multiple sources. However, our attacks can still cost time delay for synchrophasor networks (average 2 seconds). Indeed, network traffic of synchrophasor networks is time-critical. For example, the delay constraint of generic object oriented substation events (GOOSE) messages is only 4 ms in IEC 61850. Hence, with careful designs, even a weak DoS attack can cost serious consequences.



```
 27 0.000885649  48.26.87.54      4712 48.26.87.102     50984 TCP        66 3911831927  4274823301 0x8276 (3…  4712 → 50984 [SYN, ACK] Seq=3911831927 A
 28 0.000021808  48.26.87.102    50984 48.26.87.54       4712 TCP        60 4274823301  3911831928 0x61bf (2…  50984 → 4712 [ACK] Seq=4274823301 Ack=39
 29 0.000381208  48.26.87.102    50984 48.26.87.54       4712 SYNCHRO…   72 4274823301  3911831928 0x61c0 (2…  Command Frame, data transmission off
 30 0.001052108  48.26.87.54      4712 48.26.87.102     50984 TCP        60 3911831928  4274823319 0x0001 (1)  4712 → 50984 [ACK] Seq=3911831928 Ack=42
 95 0.018175587  48.26.87.102    50983 48.26.87.55       4712 SYNCHRO…   72 4161983324  1943769273 0x61da (2…  Command Frame, send CFG-2 frame
 96 0.000744937  48.26.87.55      4712 48.26.87.102     50983 TCP        60 1943769273  4161983342 0x0002 (2)  4712 → 50983 [ACK] Seq=1943769273 Ack=41
 97 0.000887219  48.26.87.55      4712 48.26.87.102     50983 SYNCHRO…  428 1943769273  4161983342 0x0003 (3)  Configuration Frame 2
 98 0.000084949  48.26.87.102    50983 48.26.87.55       4712 SYNCHRO…   72 4161983342  1943769647 0x61db (2…  Command Frame, data transmission on
 99 0.000698772  48.26.87.55      4712 48.26.87.102     50983 TCP        60 1943769647  4161983360 0x0004 (4)  4712 → 50983 [ACK] Seq=1943769647 Ack=41
102 0.000468191  48.26.87.102    50984 48.26.87.54       4712 SYNCHRO…   72 4274823319  3911831928 0x61dd (2…  Command Frame, send CFG-2 frame
103 0.000812024  48.26.87.55      4712 48.26.87.102     50983 SYNCHRO…  140 1943769647  4161983360 0x0005 (5)  Data Frame
104 0.001170071  48.26.87.54      4712 48.26.87.102     50984 TCP        60 3911831928  4274823337 0x0002 (2)  4712 → 50984 [ACK] Seq=3911831928 Ack=42
105 0.000618741  48.26.87.54      4712 48.26.87.102     50984 SYNCHRO…  428 3911831928  4274823337 0x0003 (3)  Configuration Frame 2
106 0.000114982  48.26.87.102    50984 48.26.87.54       4712 SYNCHRO…   72 4274823337  3911832302 0x61de (2…  Command Frame, data transmission on
107 0.001144187  48.26.87.54      4712 48.26.87.102     50984 TCP        60 3911832302  4274823355 0x0004 (4)  4712 → 50984 [ACK] Seq=3911832302 Ack=42
114 0.004655905  48.26.87.54      4712 48.26.87.102     50984 SYNCHRO…  140 3911832302  4274823355 0x0005 (5)  Data Frame
115 0.007079519  48.26.87.55      4712 48.26.87.102     50983 SYNCHRO…  140 1943769733  4161983360 0x0006 (6)  Data Frame
116 0.000021181  48.26.87.102    50983 48.26.87.55       4712 TCP        60 4161983360  1943769819 0x61e1 (2…  50983 → 4712 [ACK] Seq=4161983360 Ack=19
118 0.009280704  48.26.87.55      4712 48.26.87.102     50983 SYNCHRO   140 1943769819  4161983360 0x0007 (7)  Data Frame
▶ Synchronization word: 0xaa31
  Framesize: 374
  PMU/DC ID number: 4
  SOC time stamp: Oct 11, 2017 22:11:24.000000000 UTC
▶ Time quality flags
  Fraction of second (raw): 0
▼ Configuration data, 1 PMU(s) included
    Resolution of fractional second time stamp: 16777215
    Number of PMU blocks included in the frame: 1
  ▶ Station #1: "STATION 4        "
    Rate of transmission: 30 frame(s) per second
  Checksum: 0x26f1 [correct]
  [Checksum Status: Good]
```

Figure 2.7: Attacking targeting PMU. Both PMUs with IP address 48.26.87.54 and 48.26.87.55 are suffer DoS attacks that became unresponsive of select ports. After the protection mechanism is triggered, PDC then resets both PMUs by sending request CFG-2 frame.

Attacking PMUs

We also test DoS attacks targeting PMUs. In this experiment, the attacker sends a huge volume of attacking packets to a PMU. It is able to test a device's ability to create and teardown TCP sessions with floods of TCP packets targeting individual TCP ports. Our test shows that targeting devices [2] eventually stop sending synchrophasor measurements and become unresponsive when network package volume increases. Moreover, we find that local PDC employs a protection mechanism that if one PMU is unresponsive for a preset time (default 5 seconds), the local PDC will start to reset the unresponsive PMU as shown in Figure 2.7. As mentioned in Section 2.3.3, a C37.118 session begins with TCP connection handshake and follows by configuration frame request. This function is indeed beneficial for other cyber attackers because PMUs' configurations are necessary to decode C37.118 data frames.

[2] PMUs that are deployed in CURENT hardware testbed are Schweitzer Engineering Laboratories SEL-421

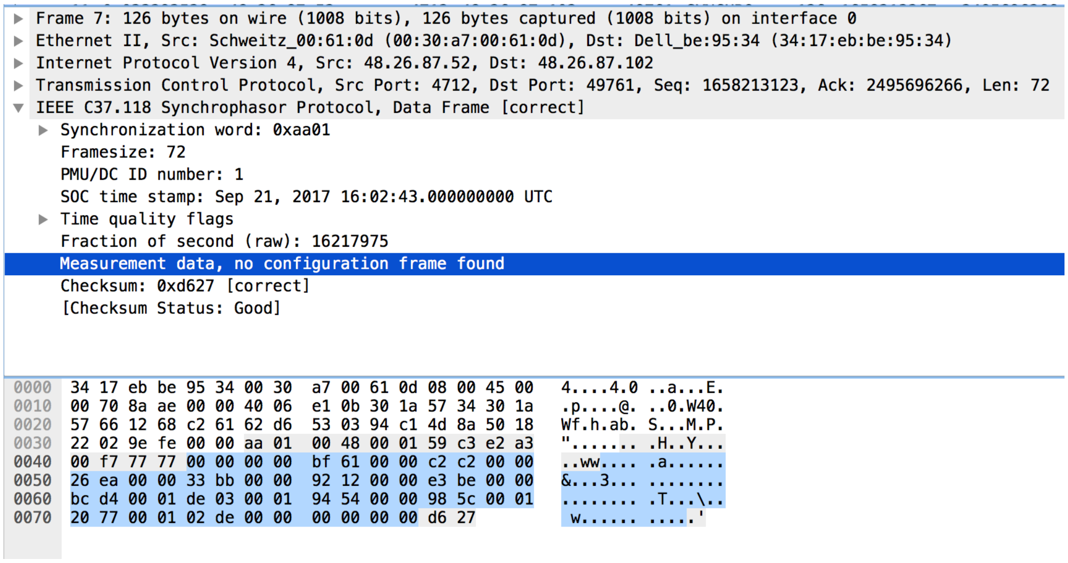
2.6.2 Sniffing Attack

A sniffing attacker can obtain sensitive information (e.g. power usage, future price information, and the smart grids' network structure) by eavesdropping network traffic which leads to privacy breaches. Sniffing attacks can be used for collecting information to perform more sophisticated attacks. For example, the adversary can monitor network traffic to deduce patterns from communication packages, and even encrypted network packages are subjected to such attacks.

For the C37.118 data frame, it is encoded by specific configuration. Each PMU's configuration is factory preset but can be reconfigured later using Telnet protocol. The configuration is crucial for data theft that without knowing PMU's configuration, it is hard to decode the data frames. Figure 2.8a shows that without the configuration frame Wireshark is unable to decode the data frame and Figure 2.8b shows that after capturing the configuration frame, Wireshark can see each phasor's voltage and current information. As mentioned in Section 2.3.3, the configuration information is exchanged when a PDC setup a connection session. It is likely that the attacker may eavesdrop for a very long period without capturing the configuration frame. However, the attacker can utilize the protection mechanism we found in the previous section to obtain the configuration frame. In this case, a sniffing attack can start with a DoS attack targeting PMU or PDC with the port that is used for sending or receiving c37.118 data frames. Under this circumstance, if the DoS attack is successful, the PDC will reset the PMU and start a new session using a different port. During the reset process, the attacker can eavesdrop the network traffic and easily capture the configuration frame.

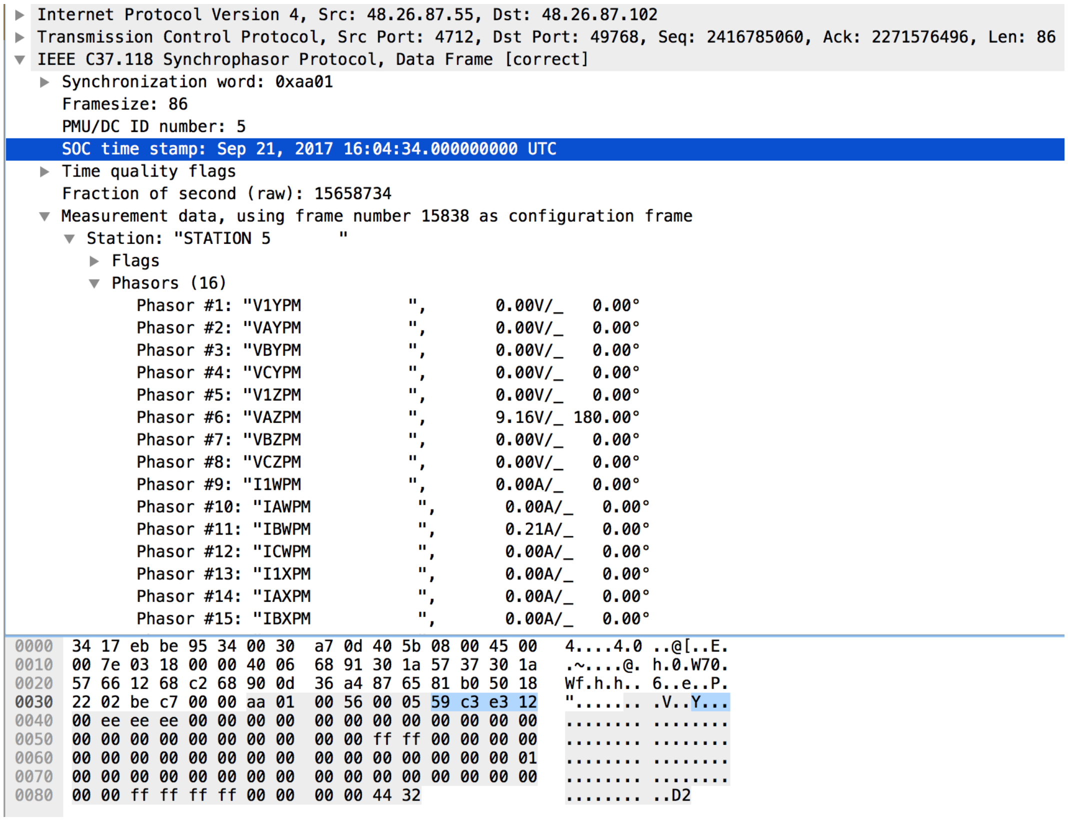
2.6.3 False Data Injection Attacks

In this section, we demonstrate a practical false data injection attack. Specifically, we show that how to start false data injection attack in the wide-area monitoring system (WAMS). We look at a scenario where a synchrophasor network is employed in a wide area to gather current, voltage and frequency measurements and to manage the situational awareness of the potential power system event.



```
▶ Frame 7: 126 bytes on wire (1008 bits), 126 bytes captured (1008 bits) on interface 0
▶ Ethernet II, Src: Schweitz_00:61:0d (00:30:a7:00:61:0d), Dst: Dell_be:95:34 (34:17:eb:be:95:34)
▶ Internet Protocol Version 4, Src: 48.26.87.52, Dst: 48.26.87.102
▶ Transmission Control Protocol, Src Port: 4712, Dst Port: 49761, Seq: 1658213123, Ack: 2495696266, Len: 72
▼ IEEE C37.118 Synchrophasor Protocol, Data Frame [correct]
  ▶ Synchronization word: 0xaa01
    Framesize: 72
    PMU/DC ID number: 1
    SOC time stamp: Sep 21, 2017 16:02:43.000000000 UTC
  ▶ Time quality flags
    Fraction of second (raw): 16217975
    Measurement data, no configuration frame found
    Checksum: 0xd627 [correct]
    [Checksum Status: Good]

0000  34 17 eb be 95 34 00 30  a7 00 61 0d 08 00 45 00   4....4.0 ..a...E.
0010  00 70 8a ae 00 00 40 06  e1 0b 30 1a 57 34 30 1a   .p....@. ..0.W40.
0020  57 66 12 68 c2 61 62 d6  53 03 94 c1 4d 8a 50 18   Wf.h.ab. S...M.P.
0030  22 02 9e fe 00 00 aa 01  00 48 00 01 59 c3 e2 a3   "....... .H..Y...
0040  00 f7 77 77 00 00 00 00  bf 61 00 00 c2 c2 00 00   ..ww.... .a......
0050  26 ea 00 00 33 bb 00 00  92 12 00 00 e3 be 00 00   &...3... ........
0060  bc d4 00 01 de 03 00 01  94 54 00 00 98 5c 00 01   ........ .T...\..
0070  20 77 00 01 02 de 00 00  00 00 00 00 d6 27          w...... .....'
```

(a) Without configuration frame



```
▶ Internet Protocol Version 4, Src: 48.26.87.55, Dst: 48.26.87.102
▶ Transmission Control Protocol, Src Port: 4712, Dst Port: 49768, Seq: 2416785060, Ack: 2271576496, Len: 86
▼ IEEE C37.118 Synchrophasor Protocol, Data Frame [correct]
  ▶ Synchronization word: 0xaa01
    Framesize: 86
    PMU/DC ID number: 5
    SOC time stamp: Sep 21, 2017 16:04:34.000000000 UTC
  ▶ Time quality flags
    Fraction of second (raw): 15658734
  ▼ Measurement data, using frame number 15838 as configuration frame
    ▼ Station: "STATION 5        "
      ▶ Flags
      ▼ Phasors (16)
          Phasor #1: "V1YPM           ",       0.00V/_   0.00°
          Phasor #2: "VAYPM           ",       0.00V/_   0.00°
          Phasor #3: "VBYPM           ",       0.00V/_   0.00°
          Phasor #4: "VCYPM           ",       0.00V/_   0.00°
          Phasor #5: "V1ZPM           ",       0.00V/_   0.00°
          Phasor #6: "VAZPM           ",       9.16V/_ 180.00°
          Phasor #7: "VBZPM           ",       0.00V/_   0.00°
          Phasor #8: "VCZPM           ",       0.00V/_   0.00°
          Phasor #9: "I1WPM           ",       0.00A/_   0.00°
          Phasor #10: "IAWPM           ",       0.00A/_   0.00°
          Phasor #11: "IBWPM           ",       0.21A/_   0.00°
          Phasor #12: "ICWPM           ",       0.00A/_   0.00°
          Phasor #13: "I1XPM           ",       0.00A/_   0.00°
          Phasor #14: "IAXPM           ",       0.00A/_   0.00°
          Phasor #15: "IBXPM           ",       0.00A/_   0.00°

0000  34 17 eb be 95 34 00 30  a7 0d 40 5b 08 00 45 00   4....4.0 ..@[..E.
0010  00 7e 03 18 00 00 40 06  68 91 30 1a 57 37 30 1a   .~....@. h.0.W70.
0020  57 66 12 68 c2 68 90 0d  36 a4 87 65 81 b0 50 18   Wf.h.h.. 6..e..P.
0030  22 02 be c7 00 00 aa 01  00 56 00 05 59 c3 e3 12   "....... .V..Y...
0040  00 ee ee ee 00 00 00 00  00 00 00 00 00 00 00 00   ........ ........
0050  00 00 00 00 00 00 00 00  00 00 ff ff 00 00 00 00   ........ ........
0060  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 01   ........ ........
0070  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ........ ........
0080  00 00 ff ff ff ff 00 00  00 00 44 32               ........ ..D2
```

(b) With configuration frame

Figure 2.8: With configuration frame, Wireshark is able to decode C37.118 data frame.

We show that a false data injection attack targeting a synchrophasor network can change the result of situational awareness and misleads the operator. We devise a false injection attack which is illustrated in Figure 2.9 and consists of the following five procedures:

1. An attacker penetrates the internal network. This process is usually done by compromising employee's email accounts, web servers, social engineering, etc.

2. The attacker eavesdrops the data communication between PDC and PMU. The data frame includes the source and destination IP addresses, port numbers, the ID of destination PDC, the message format, etc.

3. Initiate TCP flooding attack to local PDC or specific PMU. The primary purpose of this step is to affect data availability, after which PDC will reset the system by re-initiating one or multiple PMUs. Note that if the system uses UDP as the transport layer protocol, the attacker can choose other DoS attack scheme.

4. Attacker intercepts the configuration frame and uses the configuration frame to construct a false data stream. Based on the goals of the attacker, the data stream can be generated randomly or carefully manipulated using domain knowledge.

5. Wait for the good time such that the attacker can do the most damage to the power system and injects a false command frame to the PMU to shut down the data transmission. Then, the attacker sends fabricated data frames with pre-generated measurement data to local PDC to spoof the control center.

We demonstrate this attack on CURENT hardware testbed. We consider two real-world false data injection attacks: random false data injection attacks which aim to find an attack vector which can inject random errors into certain state variables, and target false data injection attacks which seek to identify an attack vector that can lead to a designated wrong state estimation result. Figure 2.10 showed the screenshot captured during a random false injection attack. After starting the attacks, the attacker takes control of the data stream transmission and keeps replying to the PDC zero value bus voltage, current, and frequency. During the attack, the visualization of synchrophasor measurements was manipulated such

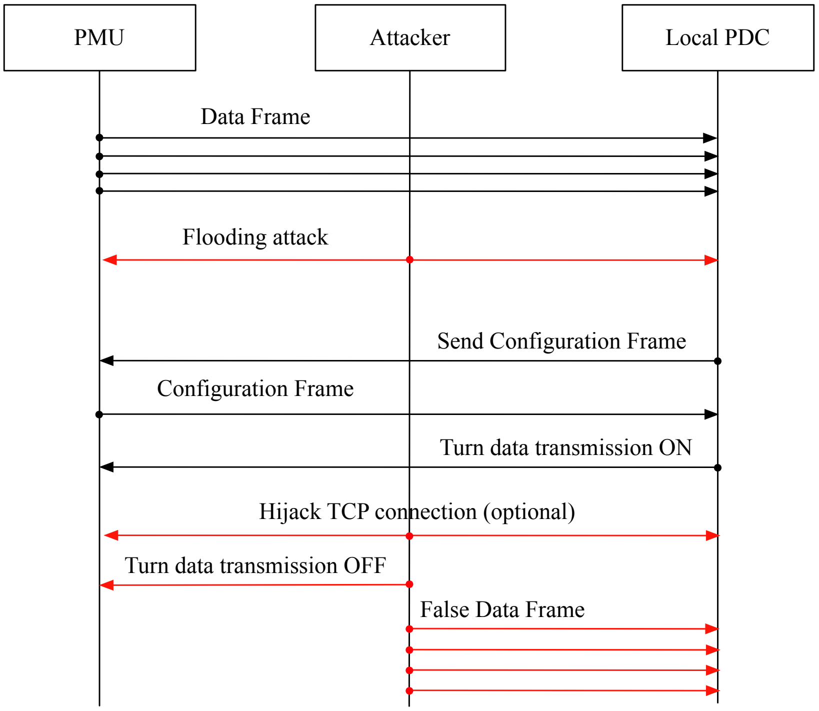
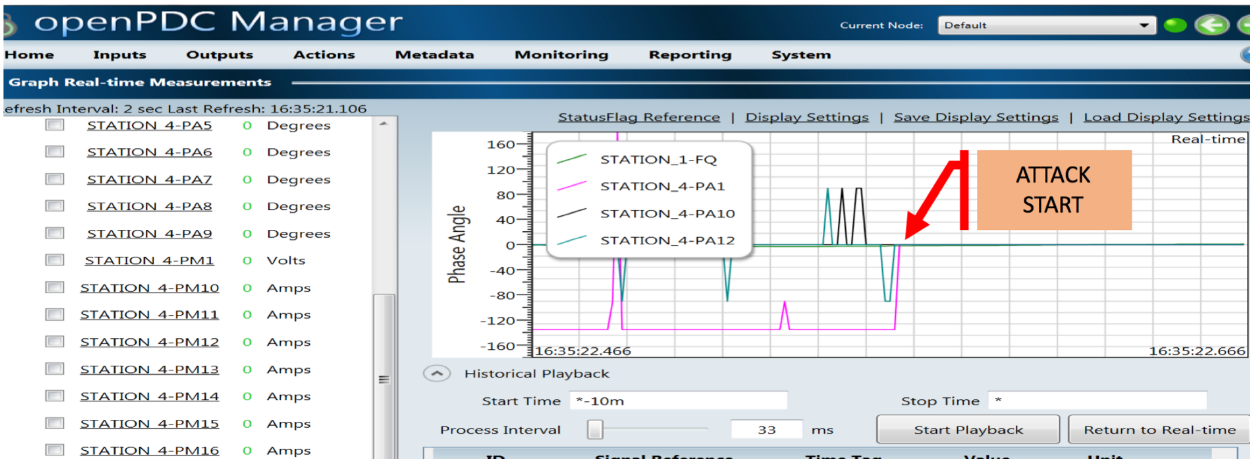


Figure 2.9: A practical false data injection attack on synchrophasor network. The black line represents the normal traffic between PMUs and PDC. And the red line denotes the attacking packets generated from the the attacker.



efresh Interval: 2 sec Last Refresh: 16:35:21.106

STATION_4-PA5	0	Degrees
STATION_4-PA6	0	Degrees
STATION_4-PA7	0	Degrees
STATION_4-PA8	0	Degrees
STATION_4-PA9	0	Degrees
STATION_4-PM1	0	Volts
STATION_4-PM10	0	Amps
STATION_4-PM11	0	Amps
STATION_4-PM12	0	Amps
STATION_4-PM13	0	Amps
STATION_4-PM14	0	Amps
STATION_4-PM15	0	Amps
STATION_4-PM16	0	Amps

Start Time *-10m

Stop Time *

Process Interval 33 ms

Figure 2.10: The demonstration of our data hijacking attack in CURENT hardware testbed.

that the control center is deceived and incapable to apply protection controls to the potential failures and oscillations.

Figure 2.11 illustrates the target false data injection attack which a generator trip event is generated to mislead the operator. This is also called replay attack where an adversary records a sequence of data measurements and replays the sequence afterward. It is worth mentioning that Stuxnet [41] was used to damage Iran's nuclear facility by inducing enormous distortions with the replay attack. For synchrophasor networks, based on this manipulated state, the operator and automatic control are blinded and harmful consequences may happen.

2.7 Attack Mitigation and Countermeasures.

In this section, we discuss possible solutions and best practices to prevent cyber attacks in synchrophasor networks. We believe that cyber-physical security can not only avoid cyber attacks but provide better solutions to detect and restore system functionalities.

2.7.1 Protecting Data Availability

First of all, it is essential to examine a device's capability of processing different network protocols which can be useful for system planning and for designing possible intrusion dection system for denial-of-service attacks. A variety of network analyzers can be operated to

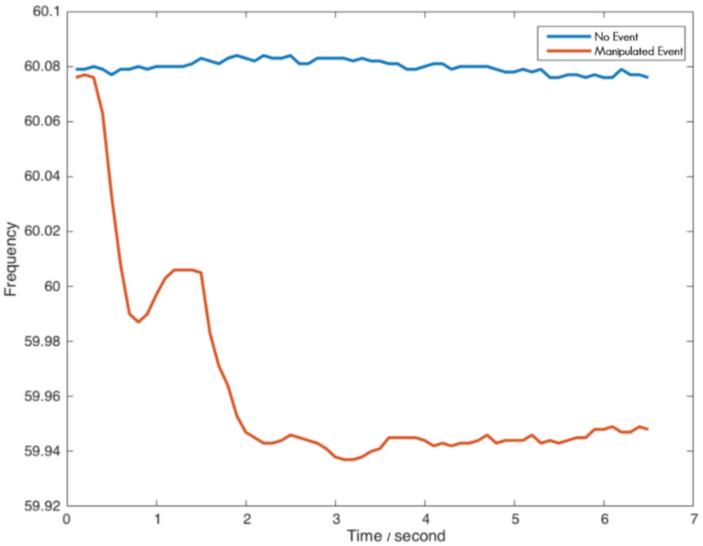


Figure 2.11: A false generator trip event is injected to CURENT hardware testbed.

perform denial-of-service attacks, network congestion and protocol mutation tests for a synchrophasor network. This step is called fuzzing or fuzz testing. Based on the fuzzing result, it is recommended that each utility company monitors network traffic and alerts administrator when the volume is over a certain threshold.

Another common practice is to employ firewalls. As a packet filter, the firewall can monitor and control inward and outward network traffic based on preset rules. Only packets that fulfill the predetermined rules can pass firewalls and anomaly packages are blocked. With a firewall developed to prevent SYN-flood, a server can resist an attack whose flooding rate is up to 22,000 SYN packets per second while an unprotected server can only defend against 500 SYN packets per second [94].

For wireless networks, to prevent an external adversary from jamming, jamming detection mechanisms can be used to identify attacks. The feasibility and effectiveness of jamming attacks have been examined in[124], and detection schemes using the MICA2 Mote platform are deployed.

The other way is using redundant devices and communication infrastructure to transmit the same measurements in separate synchrophasor networks [13]. In that case, even an attacker successfully performs DoS attacks against one synchrophasor network, WAMS can still work with redundant measurements. Thus requires attackers to compromise more devices and networks channels, which dramatically increases the complexity and cost to conduct such an attack.

2.7.2 Protecting Data Confidentiality

In general, the smart grid may contain operators, managers, engineers, etc., and these people should have different access privileges to different devices and applications by default. Using role-based access control (RBAC) can enhance the system reliability and reduce possible cyber threats. In [103], the traditional RBAC is modified in order to improve the current access management system with a higher level of granularity from the perspective of regulating power utilities.

The best way to prevent packet sniffing is to utilize SSL (Secure Socket Layer), TLS (Transport Layer Security), or IPSec (Internet Protocol Security) to encrypt the network data streams between of PDC and PMU [111]. With encryptions, most of the information, including the TCP/IP header, commands, measurements which are exchanged between PMU and PDC in the synchrophasor network, are encrypted by specific cryptography. Therefore, attackers can only get encrypted packages and obtain no valuable information through the eavesdropping without getting secret keys. However, while various secure protocols exist, some low computation capacity devices may need a lightweight protocol to meet synchrophasor network's time requirements. In Section 2.7.4, we test the overhead of deploying a particular cryptographic system.

While cryptography can considerably mitigate the risk of data breach, it is vital to avoid the key being stolen by adversaries. Key management is a necessary approach to network security. Shared secret keys and public keys can be used to decrypt secrecy and message/user authenticity. The major challenge, in this case, is key management over extensive and diverse infrastructures. While it is possible to employ traditional key exchange and distribution systems with the help of trusted third parties for the Internet, the smart grid has a wide variety of equipments and involves with different entities including government, corporations, and consumers which posts numerous additional challenges. To address these concerns, in [12], NIST provides a fundamental guideline for developing cryptographic key management systems in the smart grid.

Furthermore, for sensitive power system data that is used for specific computations, it is expected that some information will transmit across different places even with public domains. It is difficult to protect data confidentiality under this circumstance since it is possible that important data may be leaked by other parties on purpose. As a solution, secure multi-party computation aims to create methods for parties to jointly compute a function over their inputs while keeping those inputs private. In Chapter 4 and Chapter 5, we propose two privacy-preserving algorithms for optimal power flow and dynamic simulation in the multiparty scenario.

2.7.3 Protecting Data Integrity

Similar to protecting data confidentiality, cryptography plays a unique role to secure data integrity. A naive method to protect data integrity is to hash the data which you received and comparing it with the original hash. However, this means that the original hash of the data must also be kept intact against the attackers. In section 2.7.4, we test the overhead of deploying a certain authentication code system.

However, it is worth noticing that authentication is a computationally intensive procedure that can generate considerable delay and become the target of DoS attacks. For example, Fadlullah et al. [40] assume that some compromised grid devices can launch a DoS attack by frequently sending false authentication requests in the network. They propose a predication-based defending mechanism such that unusual activities such as device/authentication failures are monitored at every layer and reported to the control center.

Intrusion detection mechanisms are essential for locating compromised devices. While traditional network intrusion detection systems may be applied directly to the smart grid, several works of intrusion detection systems for the power grid have been introduced. For instance, S. Kim et al. propose to use data mining technique to learn and predict cyber attacks from synchrophasor measurements [62].

One particular cyber attack against data integrity is false data injection attack which can cost catastrophe results. For false data injection attack, bad data detector such as χ^2 or largest normalized residue detector [48] detects the corruption in measurement by checking the measurement residue. However, such a detection scheme has inherent weaknesses as

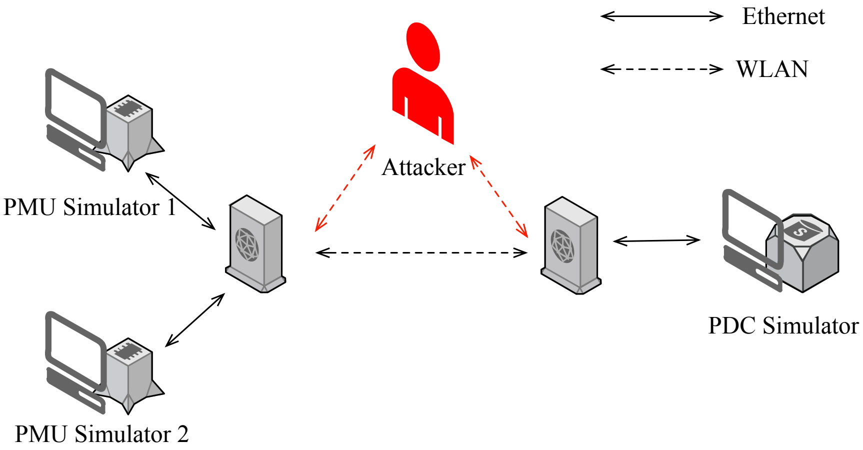


Figure 2.12: A small scaled synchrophasor network prototype.

different measurement vectors can generate the same residue. In Chapter 3, we will present a new way of detecting false data injection attack based on deep learning.

2.7.4 A Small-scale Secure Synchrophasor network

Experiment Setup

To examine protections for the synchrophasor network, we build a small-scale prototype of synchrophasor network, which is depicted in Figure 2.12. In this implementation, PDC and PMU are connected to separate gateways. The gateways have access to the wireless local area network which simulates physical distance between PMU and PDC. Additionally, since our primary concern is on the communication side, the PDC does not store synchrophasor data to a database server.

We adopt the pyPMU - Python Synchrophasor Module to emulate the PMUs and PDC with Linux operating system [107]. pyPMU is written in python and comply with the IEEE C37.118. It gets synchronized using the Network Time Protocol (NTP) protocol, and it can synthesize a sequence of random phasor measurement or generate target measurement based on needs. We also use python-cryptography as our cryptographic library which includes both high-level recipes and low-level interfaces to common cryptographic algorithms such as symmetric ciphers, message digests, and key derivation functions. It worth mention that using simulated PMU measurements does not affect the validity of this research work since we concentrate on the testing the performance of proposed communication protocol and the software implementations, which are independent of the data acquisition process.

Security Scheme

We proposed a security scheme in which PMUs and PDC perform mutual authentication at the beginning of each predefined period. Session keys are established at the end of each successful authentication and used throughout the entire period for data confidentiality and integrity. Session keys are updated with each new authentication. The proposed security scheme mitigates all the effective attacks shown in Table 2.3 and is presented as follows.

Upon receiving an authentication request from a PMU:

In this authentication process, the PDC and PMU first challenge each other by their respective nonce (a number that is used only once), $ANonce$ and $SNonce$. A session key K_{priv} is then derived from the pre-shared key (K_{pub}), the ID of the PDC (ID_{PDC}), $ANonce$, $SNonce$, and the MAC addresses of the PDC and PMU (MAC_{PDC}, MAC_{PMU}) as:

$$K_{priv} = h(K_{pub}, ID_{PDC}|ANonce|SNonce|MAC_{PDC}|MAC_{PMU}),$$

where h is a cryptographic hash function, and | denotes concatenation. The 256-bit pre-shared key should be installed into the PDC and PMU before they are shipped. Message integrity checks $MIC_{K_{priv}}(ANonce, SNonce)$ are then sent to make sure that the PDC and PMU have obtained the same session key SK. Finally, the PDC and PMU can exchange data securely by encrypting ($ENC_{K_{priv}}(Data)$) and integrity-protecting ($HMAC_{K_{priv}}(Data)$) it using K_{priv}. A good choice of the encryption algorithm is AES-CCM.

Table 2.3 shows the analytical result of the proposed security scheme. The original packet contains 16 phasors with a total size of 140 bytes. The test is conducted on a Linux machine with a 2.2 GHz CPU. We can see that AES-GCM has the best performance with minimal communication overhead. However, AES-GCM has nonce reuse problem [80] thus offers worse integrity protection than AES+HMAC. Furthermore, the performance can be future increased by integrating AES based hardware.

Table 2.3: Performance of different security protocol over IEEE C37.118

	Key Size	**Encryption Rate**	**Decryption Rate**	**Communication Overhead**
AES-GCM	256 bits	54741 packets/s	66355 packets/s	19.77%
AES-CBC+HMAC-SHA256	256 bits	51503 packets/s	50495 packets/s	76.74%
ChaCha20+HMAC-SHA256	256 bits	41511 packets/s	40198 packets/s	86.04%

Chapter 3

Dynamic Detection of False Data Injection Attack in Smart Grid using Deep Learning

3.1 Introduction

The future smart grid is designed to operate more reliable, economical and efficient in an environment of increasing power demand. This goal, however, is achieved by incorporating with a tremendous increase of data communications which lead to great opportunities for a variety of cyber attacks. Thus, ensuring cybersecurity of the smart grid is a critical priority. Although a large number of countermeasures have been published, such as communication standards (e.g. IEC 61850-90-5 [56]), regulation laws (e.g. Colorado Regulations (CCR) 723-3), cryptographic implementations (e.g. secure channel [43]), and official guidelines (e.g. NISTIR 7628 Guidelines [91]), current smart grid still remains vulnerable to cyber attacks.

To prevent cyber attacks, legacy grid relies on the traditional security scheme (e.g., firewall and general intrusion detection system). Intrusion detection systems (IDS) can generate alarms for potential intrusions by consistently monitoring network traffic or system logs. Although there are a number of studies on general IDS in the network security community, limited effort has been explicitly made to the smart grid. At the same time, the cyber threat against data integrity in the power system is indeed real. For instance, Y. Liu et al. [75] proposed a type of attacks targeting the state estimation in the electric power grid called false data injection (FDI) attacks. In such attacks, the attackers aim to bypass existing bad data detection system and pose damage to the operation of the power system by intentional changing the estimated state of the grid systems. Z. Chu et al. [30] introduce four computationally efficient algorithms to evaluate the vulnerability of large-scale power systems to FDI attacks and found that all critical lines are vulnerable. Therefore, there is an urgent need of effective smart grid specific intrusion detection systems.

Recently, machine learning algorithms have been broadly adopted to the smart grid literature for monitoring and preventing cyber attacks on power systems. Ozay et al. [95] generate Gaussian distributed attacks and use both supervised and semi-supervised machine learning methods to classify attacks. Similarly, Esmalifalak et al. [38] devise a distributed support vector machines based model for labeled data and a statistical anomaly detector for unsupervised learning cases. He et al. [53] utilize Conditional Deep Belief Network (CDBN) to discover the high-dimensional features between normal data and unobservable FDI attacks. However, existing works mainly focus on finding bad measurement at a specific state, no prior studies have been conducted over the dynamic behavior of FDI attack. Besides, detecting FDI attacks are considered as supervised binary classification problem in [38, 53] which are incapable of identifying dynamically evolving cyber threats and changing system configuration.

The recent breakthrough in GPU computing provides the foundation for neural networks to go ”deep.” In this chapter, we develop an anomaly detection framework based on neural networks to enable the construction of a smart grid specific IDS. More specifically, a recurrent neural network with LSTM [54] cell is deployed to capture the dynamic behavior of the power system and a convolutional neural network [65] is adopt to balance between two input sources. An attack is alerted when the residual between the observed and the estimated measurements is greater than a given threshold.

Moreover, attackers with sophistic domain knowledge may continually manipulate the power grid state estimation without being detected causing extensive damages. As such, we want to bridge the gap between network anomaly detector and FDI attacks detection mechanism. Unlike other works which separate two detectors, our framework combines both network traffic characteristics and time-series data measurements with the help of convolution neural network to equalize between two inputs. With the help of the proposed neural network structure, our anomaly detector demonstrates highly accurate detection performance.

3.1.1 Challenges

Recent developments in neural networks have dramatically improved machine learning technology. This advancement highlights the potentials for many applications that aim to build systems that can identify patterns and make decisions from data with minimal human intervention. For information security, we also observed a significant number of new works that utilize deep learning for a diversity of applications which also includes frameworks for anomaly detection.

In the meantime, although prior work already applied supervised machine learning for detecting FDI attacks in [95], it is not trivial to find the best machine learning algorithm for the dynamic environment in the smart grid. First, despite that Y. Liu already brought out basic principle and some scenarios for FDI attacks in [75], it is still unlikely that we can provide a thorough dataset that contains all possible attacking schemes. Consequently, the accuracy of the trained model will greatly decrease when the attacking vector is different from the training set. In addition, the new powerful malicious computer worm targeting cyber-physical systems provided the opportunities for the attackers to carry out more powerful attacks such as replay attack. More specifically, it is feasible for the attackers to compromise a large set of PMUs or PDCs in synchrophasor networks and inject phasor measurements that are captured from a real event in order to mislead the power system operators (see Section 2.6.3 for how to start FDI attacks).

Furthermore, substations often employ network IDS to detect network packages anomalies, however, often run separately from FDI attacks detection system. Apparently, running two independent system is less efficient and can increase false alarm rate. It is difficult to create a unified model to combine both systems because network IDS takes dynamic network flows as input and FDI attacks detectors are usually based on static vector estimation.

By dealing with the aforementioned challenges, we are motivated to develop a dynamic FDI attack detection method that takes advantage of both network features and data measurements.

3.1.2 Outlines

We organize the rest of this chapter as follows: We first review some millstone works for FDI attacks in Section 3.2. Section 3.3 introduces the background of FDI attack and neural networks. Section 3.4 presents our combined detection system along with the static and dynamic method to detect FDI attack. Section 3.5 validate the proposed system with a case study on IEEE 10-machine 39-bus power system.

3.2 Related Work

After FDI attack was first introduced in 2009 [75], numerous works on composing and defending against FDI attacks have been proposed in the past decade. To address the above issues, two schemes have been widely studied to defend FDI attacks [19, 71]: One way is to protect a number of secure basic measurements strategically. Kim et al. [63] propose a greedy algorithm to select a subset of base measurements and the placement of secure phasor measurement units. Bi et al. [17] characterize the problem into a graphical defending mechanism to select the minimum number of meter measurements which cannot be compromised.

The other way of defending FDI attack is to verify each state variables independently. Liu et al. [73] formulate a low-rank matrix separation problem to identify attacks and propose two optimization methods to solve the problem. Ashok et al. [9] present an online detection algorithm that utilizes statistical information and the predictions of the state variables to detect measurement anomalies. Yang et al. [125] proposed efficient algorithms to identify the optimal attacking meter set. A protection-based defense scheme and a detection-based defense scheme are introduced to defend such attacks. In [78], the authors adopt the Kalman filter to estimate the state variables and the outputs are fed into a χ^2-detector or a Euclidean detector. What's more, some researches have been carried out to form electricity thieves using methods such as game theory. Cárdenas et al. [26] formulate a game between the electric utility and the electricity thief where the thief aims at stealing a fixed amount of energy while minimizing the likelihood of being detected and the utility need to determine the tradeoff between the probability of detection and the cost of the detection system. Some other methods have been proposed to detect the electricity theft by analyzing the abnormal consumption patterns [86, 87]

Recently, with the fast development in machine learning technologies, many machine learning techniques have been applied to develop an anomaly-based intrusion detection system for cyber-physical system. Specifically, machine learning can automatically learn patterns and make predictions from the data. For smart grid, these anomaly-based intrusion detection systems employ SCADA or synchrophasor networks to create normal behavior of current power system, then detect anomalies which are different with the learned patterns, and thus capable to detect unseen attacks. For example, Ozay et al. [95] reformulate the FDI attacks detection problem as a machine learning problem and state-of-the-art machine learning algorithms are examined in this scenario. He et al. [53] utilize Conditional Deep Belief Network (CDBN) to discover the high-dimensional features between normal data and unobservable FDI attacks. However, those methods still focus on detecting malicious data based on the static state.

3.3 Preliminary

In this section, we briefly review false data injection (FDI) attack and basic neural networks structures that are used to detect FDI attacks.

3.3.1 False Data Injection Attack

State Estimation in Power Systems

In a power system, the state is represented by bus voltage magnitudes $V \in \mathcal{R}^n$ and angles $\theta \in ([-\pi, \pi])^n$, where n is the number of buses. Let $z = [z_1, z_2, ..., z_m]^T \in \mathcal{R}^m$ be the measurement vector, $x = [x_1, x_2, ..., x_n]^T \in \mathcal{R}^n$ be the state vector, and $e = [e_1, e_2, ..., e_m]^T \in \mathcal{R}^m$ denote the measurement error vector. We describe the AC measurement model as follows:

$$z = h(x) + e \tag{3.1}$$

To analyze the impact of FDI attack on state estimation, we choose to adopt the DC model which is the linearization of the AC model. For DC state estimation, the relationship between these m meter measurements and n state variables can be expressed as a $m \times n$ matrix $\mathbf{H}$. Generally, for a given power system, the matrix $\mathbf{H}$ is a constant matrix which is determined by the line impedances and network topology. An observable power network is when there are enough measurements to ensure the state estimation of the current network.

$$z = \mathbf{H}x + e \tag{3.2}$$

Typically, a weighted Least Square Estimation (LSE) is used to obtain the state estimate as:

$$\begin{aligned} \hat{x} &= \min_x \tfrac{1}{2}(z - \mathbf{H}x)^T \mathbf{R}^{-1}(z - \mathbf{H}x) \\ &= (\mathbf{H}^T \mathbf{R}^{-1} \mathbf{H})^{-1} \mathbf{H}^T \mathbf{R}^{-1} z \end{aligned} \tag{3.3}$$

where $\mathbf{R}$ is the covariance matrix and its elements are reciprocals of the meter errors variances. To solve 3.3, QR decomposition method can be leveraged to avoid calculating matrix inverse.

Bad Data Detector

For detecting a false data injection attack, one method is to use a traditional statistical test based on the degrees of freedom (redundancy) in the system. It can be shown that under Gaussian noise, the total normalized square measurement error, $J(\hat{x})$ follows a chi-square distribution based on the measurement redundancy with $(m - n)$ degrees of freedom. The test for bad measurement is then:

$$J(\hat{x}) < \tau \tag{3.4}$$

If $J(\hat{x}) > \tau$, bad data will be suspected with τ obtained from the χ^2 distribution over $(m-n)$. Once bad data is suspected, the bad needs to be identified and corrected if possible. Bad data are assumed to be those with the largest normalized residual (LNR). The LNR test selects the largest the $l2$-norm of the measurement residual [48] as the most likely suspect measurement.

For DC model, the traditional bad data detection approaches often reduce to $l2$-norm of the measurement residual [75]:

$$\begin{aligned} &\|\mathbf{H}x + e\|_2 \leq \tau \\ &\|\mathbf{H}x + e\|_2 > \tau \end{aligned} \tag{3.5}$$

However, this method can be easy mitigated by choosing a as a linear combination of the column vectors of H. Which means, if the attacker is able to use Hc as the attack vector a, z_a can pass the detection as long as regular measurement z can pass the detection.

Attack Models

Let z_a denotes the vector of actual measurements that may contain malicious injections. z_a can be represented as $z_a = z + a$ where $a = (a_1, ..., a_m)^T$ is the malicious data added to the original measurements. Let x_a represents the estimates of x using the malicious measurements z_a. Then x_a can be expressed as $\hat{x}+c$, where c is a non-zero vector representing the impact on the estimate from the malicious injection and $\hat{x}$ is the estimate using the original measurements. The $l2$-norm of the measurement z_a residual can be computed as:

$$\begin{aligned} \|z_a - \mathbf{H}x\|_2 &= \|z + a - \mathbf{H}(x + c)\|_2 \\ &= \|z + a - \mathbf{H}x - \mathbf{H}c)\|_2 \\ &= \|z - \mathbf{H}x\|_2 + \|a - \mathbf{H}c\|_2 \leq \tau \end{aligned} \tag{3.6}$$

In this work, for target FDI attackers, we assume the attacker has enough inside information to construct x_a while random FDI attackers only have partial information. Under FDI attacks, the system operators are misled by considering the biased estimations x_a as the correct value of the current state. In this case, FDI attacks are also called "unobservable" attacks.

3.3.2 Convolutional Neural Network

Convolutional Neural Networks (CNNs or ConvNets) are a family of Neural Networks that is shown in Figure 3.2. CNNs have been successful in areas such as image recognition and classification in identifying faces and objects. CNNs are special variants of multi-layer perceptron (MLP). The architect of a typical MLP is illustrated in Figure 3.1. Formally, a one-hidden-layer MLP is a function:

$$f(x) = G(b^{(1,t)} + W^{(1,t)}x) \tag{3.7}$$

where G is the activation function; $W^{(t)}$ and $b^{(t)}$ denote the weight matrices and the bias vectors. The output is then obtained as: $o(x) = G(b^{(2,t)} + W^{(2,t)}h(x))$

CNNs describe the most classic form of neural network where multiple processing nodes are arranged in layers such that information only flows from input to output. We mainly use three types of layers to build a CNN architecture: Convolutional Layer, Pooling Layer, and Fully-Connected Layer.

- Convolutional layer preserves the spatial relationship by applying a convolution operation, which computing a dot product between their weights and followed by a non-linear function where we often use rectified linear unit (ReLU): $f(z) = max(0, z)$.

- Pooling layer tries to reduce the dimensionality of the input but preserves the most important information. Pooling layer can be of different types: max pooling, average pooling, sum pooling, etc.

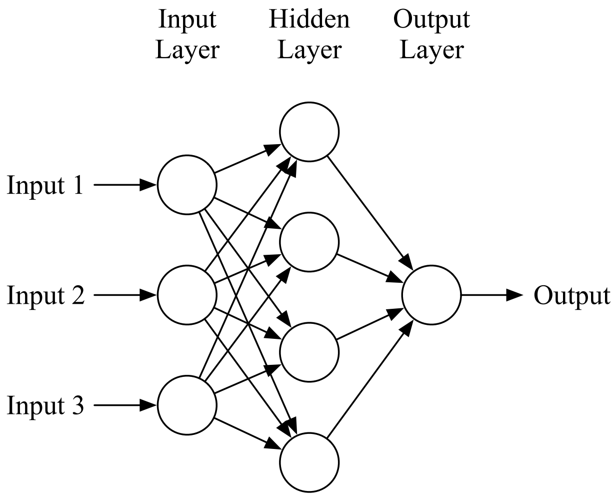


Figure 3.1: A MLP with one hidden layer.

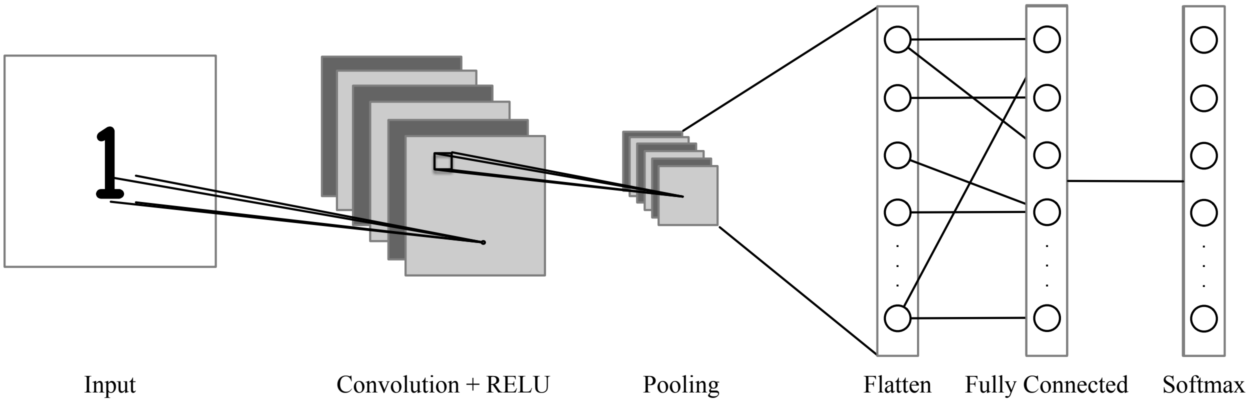


Figure 3.2: An example of architecture for classification with convolutional neural network.

- The fully connected layer is a one-hidden-layer MLP that uses a softmax activation function (multi-classification task) or sigmoid activation function (binary classification task) in the output layer.

CNN performs transformations to the original input into the final class values. Note that some layers contain parameters that need to be learned, and others don't. In particular, the convolutional layers and fully connected layers can be expressed as a function that not only contains the weights and biases parameters but also the non-linear activations. The parameters in the CNN can be trained with optimization algorithms, such as gradient descent, that the output scores are consistent with each input's label in the training set.

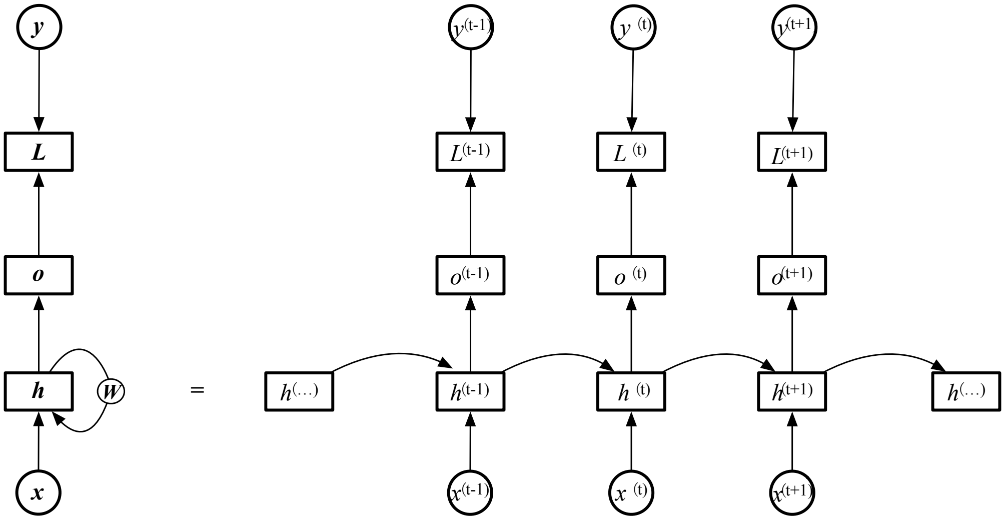


Figure 3.3: A recurrent neural network with no outputs.

3.3.3 Recurrent Neural Network

Recurrent neural networks, or RNNs [105], are a category of neural networks that are specialized for processing sequential data. While a convolutional network is good at processing tensors such as an image, a recurrent neural network is specialized for handling variable-length sequence input $x^{(1)}, ..., x^{(t)}$. Each layer performs the same task without sequence-based specialization, and it can scale to much longer sequences than other methods.

The hidden units of recurrent neural networks are defined in (3.8) or a similar equation.

$$h^{(t)} = f(h^{(t-1)}, x^{(t)}; \theta), \tag{3.8}$$

where h represent the state and $x^{(t)}$ refers to the recurrent input at time t. This equation can be described by a directed computational graph as illustrated in Figure 3.3.

The recurrent neural network is usually trained to predict future sequence from the history, the RNN uses $h^{(t)}$ as a loss function over y with respect to the past input sequence up to t. Given its current $h^{(t)}$, the RNN outputs a probability distribution over an arbitrary length sequence $(x^{(t)}, x^{(t-1)}, ..., x^{(1)})$. Unfortunately, Bengio et al. [15] observes that it is difficult to train RNNs with long-term dependencies due to common gradient vanish or explode problems. We may be able to initialize the weights very carefully to avoid gradient vanish or explode problems, however, it's still hard to capture long-time the dependency.

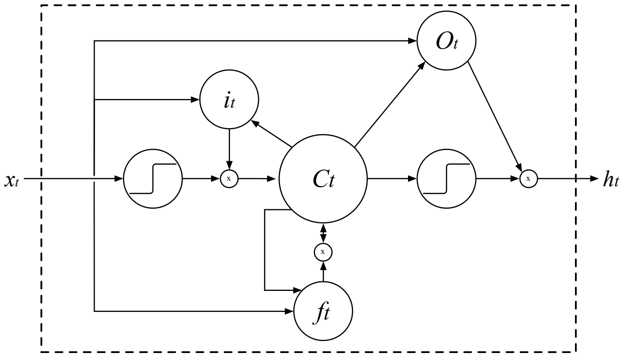


Figure 3.4: The block diagram of the LSTM recurrent network.

Long Short-Time Memory

The most commonly implemented RNNs fall into the class of long short-time memory (LSTM [54]) neural networks. As the name suggests, such RNNs exhibit remarkable empirical performance for extracting/preserving long-term dependencies whilst also maintaining short-term signals. LSTM networks, as shown in Figure 3.4, involve three gates in the computation of each hidden cell to determine what to forget, what to output and what to be provided to next hidden cell respectively. The information flow of LSTM cell is as follows:

$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f) \tag{3.9}$$

$$i_t = \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \tag{3.10}$$

$$o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \tag{3.11}$$

$$c_t = f_t \circ c_{t-1} + i_t \circ \sigma_c(W_c x_t + U_c h_{t-1} + b_c) \tag{3.12}$$

$$h_t = o_t \circ \sigma_h(c_t) \tag{3.13}$$

where $\sigma_g(\cdot)$ and $\sigma_c(\cdot)$ represent the sigmoid and tangent function, respectively, and $\circ$ denotes the element-wise product.

Bidirectional RNN

While conventional RNNs provide a very efficient way of dealing with sequential data, it only exams past information to predict the next data. As such, Schuster et al. [108] proposed a bidirectional RNN neural network (Bi-directional RNNs) that makes predictions based

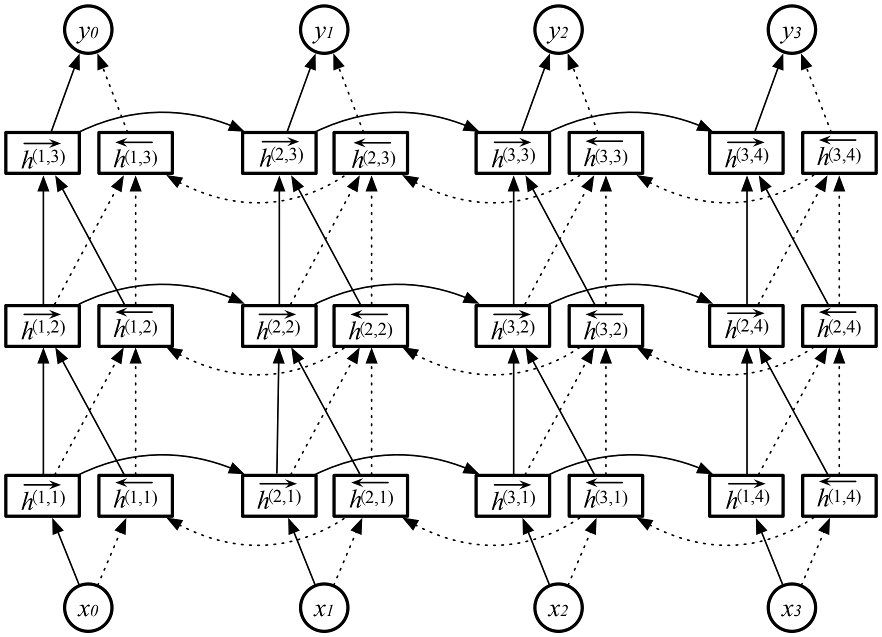
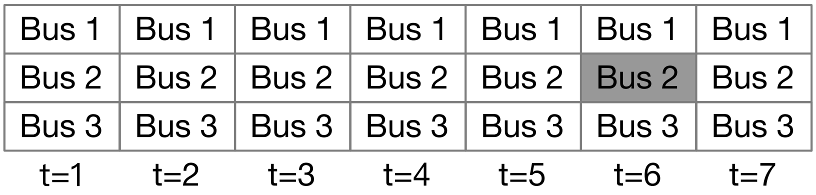

y0
y1
y2
y3
$\overrightarrow{h}^{(1,3)}$
$\overleftarrow{h}^{(1,3)}$
$\overrightarrow{h}^{(2,3)}$
$\overleftarrow{h}^{(2,3)}$
$\overrightarrow{h}^{(3,3)}$
$\overleftarrow{h}^{(3,3)}$
$\overrightarrow{h}^{(3,4)}$
$\overleftarrow{h}^{(3,4)}$
$\overrightarrow{h}^{(1,2)}$
$\overleftarrow{h}^{(1,2)}$
$\overrightarrow{h}^{(2,2)}$
$\overleftarrow{h}^{(2,2)}$
$\overrightarrow{h}^{(3,2)}$
$\overleftarrow{h}^{(3,2)}$
$\overrightarrow{h}^{(2,4)}$
$\overleftarrow{h}^{(2,4)}$
$\overrightarrow{h}^{(1,1)}$
$\overleftarrow{h}^{(1,1)}$
$\overrightarrow{h}^{(2,1)}$
$\overleftarrow{h}^{(2,1)}$
$\overrightarrow{h}^{(3,1)}$
$\overleftarrow{h}^{(3,1)}$
$\overrightarrow{h}^{(1,4)}$
$\overleftarrow{h}^{(1,4)}$
x0
x1
x2
x3

Figure 3.5: Computation of a typical 3 layers bidirectional recurrent neural network.

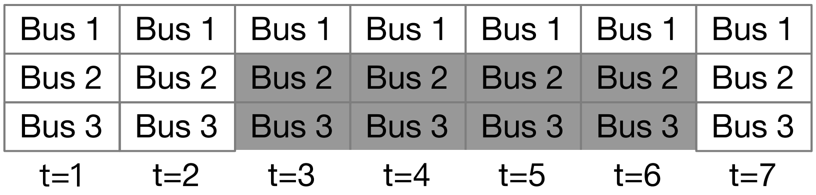
on both past and future sequences by adding another hidden layer. At each time-step t, bidirectional RNN is split to two hidden layers for each neuron, one is used for the forward propagation and another is used for backward propagation. The final class score, y^t, is calculated by merging the two scores generated by both forward and backward hidden layers. This network, however, consumes twice as much memory space due to the increased number of hidden layers. Figure 3.5 shows the bi-directional network architecture, and (3.14) presents the mathematical formulation of bidirectional RNN. The difference between these traditional RNN and bidirectional RNN is the direction of recursing through the "cells."

$$\overrightarrow{h}(t) = f(h^{(t-1)}, x^{(t)}; \overrightarrow{\theta}) \tag{3.14}$$

$$\overleftarrow{h}(t) = f(h^{(t+1)}, x^{(t)}; \overleftarrow{\theta}) \tag{3.15}$$


Bus 1 Bus 1 Bus 1 Bus 1 Bus 1 Bus 1 Bus 1
Bus 2 Bus 2 Bus 2 Bus 2 Bus 2 Bus 2 Bus 2
Bus 3 Bus 3 Bus 3 Bus 3 Bus 3 Bus 3 Bus 3
t=1 t=2 t=3 t=4 t=5 t=6 t=7

(a) Limited power FDI attacks


Bus 1 Bus 1 Bus 1 Bus 1 Bus 1 Bus 1 Bus 1
Bus 2 Bus 2 Bus 2 Bus 2 Bus 2 Bus 2 Bus 2
Bus 3 Bus 3 Bus 3 Bus 3 Bus 3 Bus 3 Bus 3
t=1 t=2 t=3 t=4 t=5 t=6 t=7

(b) Unlimited power FDI attacks

Figure 3.6: Different scenario for dynamic FDI attacks.

3.4 Real-time Detection of False Data Injection Attack

Various research on static FDI attack detection method has been published. A common assumption is a threat model where the attackers have knowledge of the power system topology; however, can only inject a limited number of bad data points which is shown in Figure. 3.6a. In this threat model, FDI attack can be mitigated if a proportion of the comprised substation is below a certain threshold. Moreover, data measurements are often redundant for estimating the actual state. This threat model is widely adopted in existing works. Nonetheless, we stress this threat mode by 1) removing the limitation of the number of measurement data that are corrupted; and 2) assuming the attackers have a basic understanding of the aforementioned static detection mechanism in (3.5).

Figure 3.6b shows the dynamic FDI attack that is focused on this work. The attack starts at $t = 3$, and the measurements of both bus 2 and 3 have been compromised. Static methods may fail in this scenario, for the reason that two-thirds of the measurements have been modified from $t = 3$ to $t = 6$. A sophisticated attacker can deliberately generate a false event based on a real event and inject it into the power grid. As a result, it is unlikely to detect this attack only based on static methods which can have catastrophic consequences if the control center takes false actions.

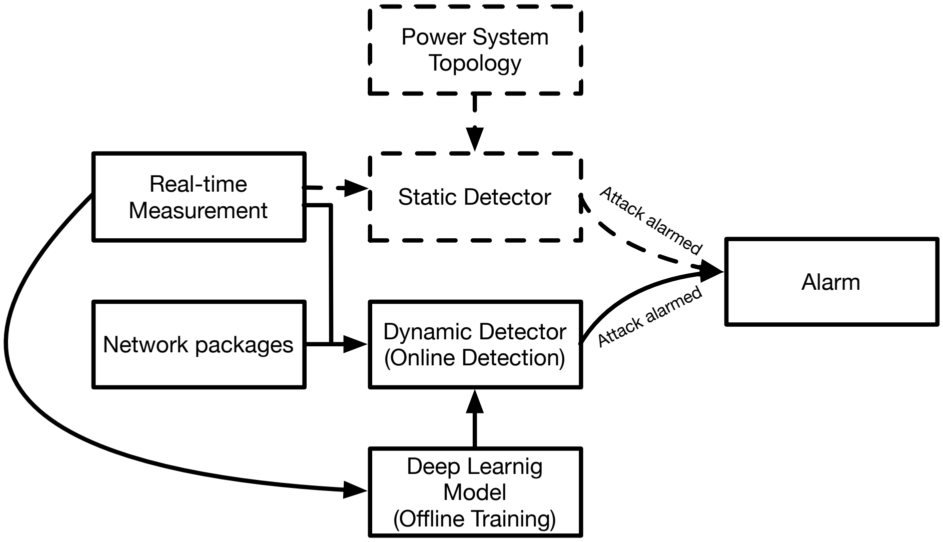


Figure 3.7: The overview of our proposed deep learning based FDI attacks detection system.

3.4.1 The Combined Attack Detection Method

In this section, we provide an overview of our proposed system for detecting FDI attacks in Figure 3.7. Our proposed detection mechanism mainly consists of a static detector and a deep learning based dynamic detection scheme. The static detector can be a State Estimator (SE) or any aforementioned FDI attack detector [9, 17, 38, 53, 73, 95, 63] which is built independently beyond our dynamic detector. As mentioned in the previous section, the dynamic detector takes two input sources. While the data level features are explicit, the network packages are captured by tcpdump, and each network packet includes header and data payload, with unique features which defined in NSL-KDD dataset [114]. The NSL-KDD dataset has 41 features as shown in Table 3.1 which are categorized into three types: basic, content-based, and traffic-based features. It should also be mentioned that some features are generated based on a fixed window (default is 2 second) which will remain consistent within the window.

Our dynamic detector is employed to recognize the high-level time-series features of the FDI attacks. To achieve this goal, our time-series method consists of two essential mechanisms: offline training and online detection. The offline training is trained based on historical measurement and can be potentially facilitated by outsourcing to public machine learning cloud services. Unlike other methods which are designed under the assumption that

Table 3.1: Derived features that are extracted from network trace

Feature Name	**Description**	**Type**
duration	length (number of seconds) of the connection	continuous
protocol_type	type of the protocol, e.g. tcp, udp, etc.	discrete
service	network service on the destination, e.g., http, telnet, etc.	discrete
src_bytes	number of data bytes from source to destination	continuous
dst_bytes	number of data bytes from destination to source	continuous
flag	normal or error status of the connection	discrete
land	1 if connection is from/to the same host/port; 0 otherwise	discrete
wrong_fragment	number of “wrong” fragments	continuous
urgent	number of urgent packets	continuous
hot	number of “hot” indicators	continuous
num_failed_logins	number of failed login attempts	continuous
logged_in	1 if successfully logged in; 0 otherwise	discrete
num_compromised	number of “compromised” conditions	continuous
root_shell	1 if root shell is obtained; 0 otherwise	discrete
su_attempted	1 if “su root” command attempted; 0 otherwise	discrete
num_root	number of “root” accesses	continuous
num_file_creations	number of file creation operations	continuous
num_shells	number of shell prompts	continuous
num_access_files	number of operations on access control files	continuous
num_outbound_cmds	number of outbound commands in an ftp session	continuous
is_hot_login	1 if the login belongs to the “hot” list; 0 otherwise	discrete
is_guest_login	1 if the login is a “guest”login; 0 otherwise	discrete
count	number of connections to the same host as the current connection in the past two seconds	continuous
serror_rate	% of connections that have “SYN” errors	continuous
rerror_rate	% of connections that have “REJ” errors	continuous
same_srv_rate	% of connections to the same service	continuous
diff_srv_rate	% of connections to different services	continuous
srv_count	number of connections to the same service as the current connection in the past two seconds	continuous
srv_serror_rate	% of connections that have “SYN” errors	continuous
srv_rerror_rate	% of connections that have “REJ” errors	continuous
srv_diff_host_rate	% of connections to different hosts	continuous

the physical status of the power system does not change overtimes, our system will collect real-time measurement data to support offline training and the prediction model will update after retrain is completed.

3.4.2 Static Detection Method

In general, if the malicious data a is unstructured, the attack vector is likely to be detected by SVE. So we can use a simple SVE to detect some limited power FDI attacks.

3.4.3 Dynamic Detection Method

In [52], the authors formulate the bus voltage magnitudes, angles and states of measuring devices together as system states in Markov Decision Process (MDPs). In our method, we extend to a recursive model where the decision not only depends previous one state but previous n states where the loss is as follows:

$$\eta = L(\phi(s_t), f(\phi(s_{t-1}, ..., s_{t-n-1}), \theta), \tau) \tag{3.16}$$

where ϕ, θ are parameters need to be turned and τ is the threshold that is needed to decide whether the attack has been started.

Figure 3.8 shows the architecture of our dynamic detection method. More explicitly, the input of the model is the time-serious power system data, and the features will be transformed by several bi-directional RNN layers to learn high-dimensional temporal features. Previous works [42, 53] characterize FDI attacks as a binary classification problem which looks promising in the experimental setting, since the datasets to be tested can be manually tuned for different scenarios. In real-world implementations, power system data is highly unbalanced. Thus, binary classification methods will inevitably have low recall even the overall accuracy is high. However, for evaluating IDS, recall is often more important than accuracy since any cyber attack can cost catastrophe results.

In general, our dynamic anomaly detector takes the time-series input $..., x^{(t-1)}, x^{(t+1)}$ to learn their high-dimensional feature representations, and then use learned representations to

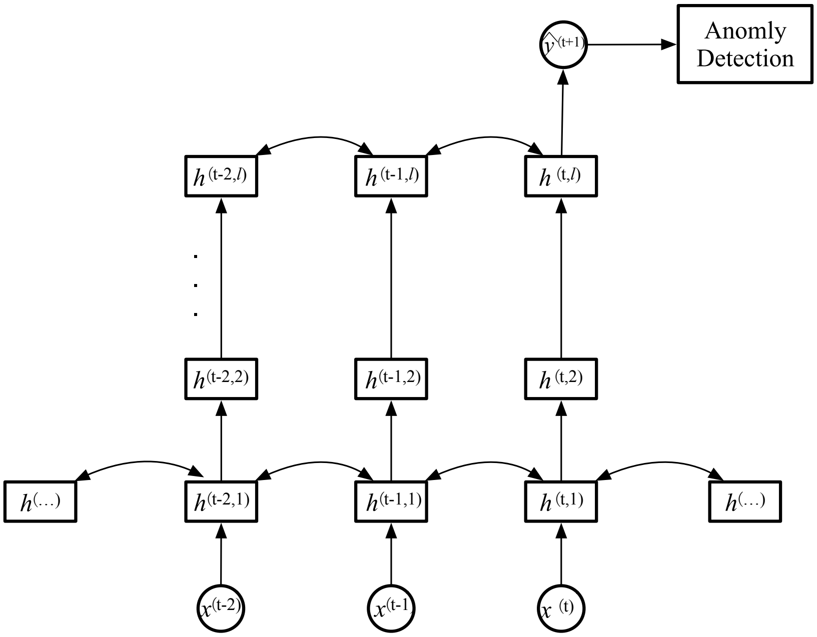


Figure 3.8: The time-serious dynamic detection method based on RNN.

predict data $\hat{x}^{(t)}$. After that, we use the predicted data to classify if $x(t)$ is anomalous by calculating the residue between the actual data $x(t)$ and predicted data $\hat{x}^{(t)}$.

Having presented single source FDI attacks detection model, we now introduce a framework that combines FDI attacks detector with network intrusion detection system. This framework is dealing with a case when an IDS that relies on data measurement fails to detect the start of FDI attacks. Accordingly, if the fabricated injection data are derived from a legitimate measurement in our threat model, data level detectors may fail to determine if the current network is intruded or not. In this case, to increase the overall performance of the time-series anomaly detection model, a combined attack detection method is proposed in this research effort.

Specifically, the schematic structure of the proposed scheme is given in Figure 3.9. An alternative method to combine data level information and packet level features is directly concatenated with the input vector. However, because the dimension between the data measurements and network packet features differ significantly, direct concatenation may have minimal improvement than aforementioned time-serious methods. Alternatively, each level features are transformed by a convolutional neural network before concatenation as shown in the figure. The purpose of adding additional convolutional neural network is to equalize the

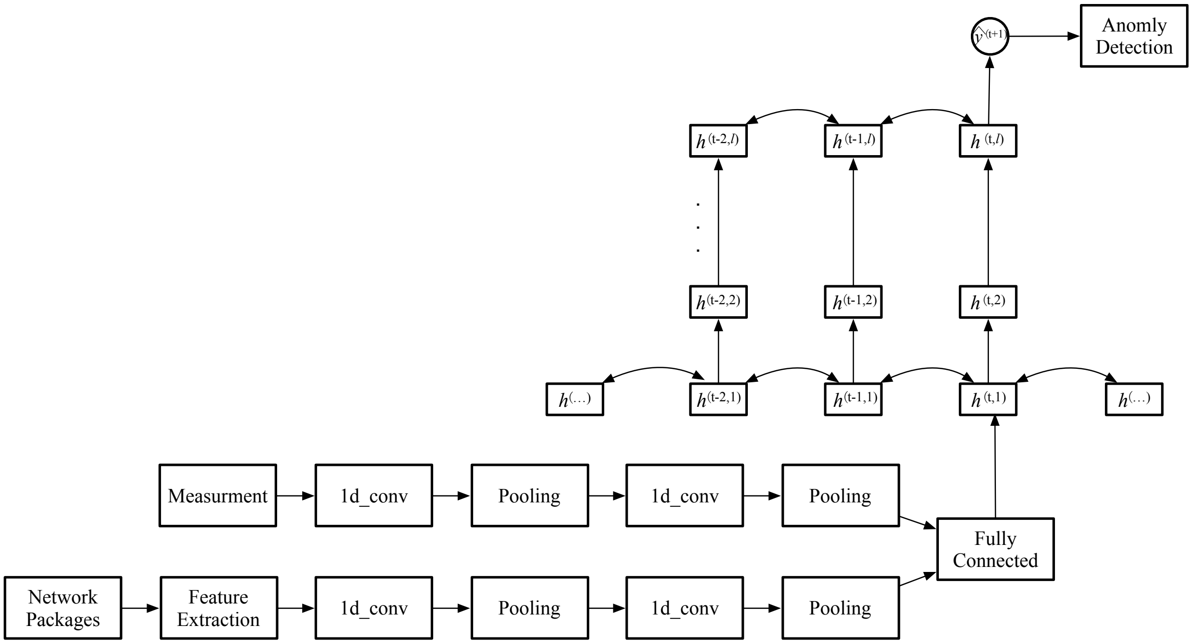


Figure 3.9: The combined detection method with both network package and data measurement inputs.

dimension between data measurement and packet level features and their respective weights are learned using gradient descent (Adam algorithm is used in our experiments). Inception deep learning architecture [113] is advised when possible.

3.5 Evaluation

In this section, we provide several key implementation details of our proposed FDI attack detection system, thereby providing a better intuition about its capabilities and limitations. Figure 3.10 shows IEEE 10 generator 39 bus power system and details in [96]. In the 39 bus system, the state vector $x \in \mathbb{R}^{39}$ is composed of the voltage, current, and frequency of the individual buses. The communication network is emulated using two computers where one computer represents the Independent Service Operator which collect data measurement through Ethernet. The sample rate is set to 10Hz. The FDI attacks are generated from man-in-middle attackers from a client-server communication structure, and two input sources are time synchronized to make it possible for real-time implementation. The dynamic detector is configured with three layers bi-directional RNN with LSTM cells and trained using Pytorch.

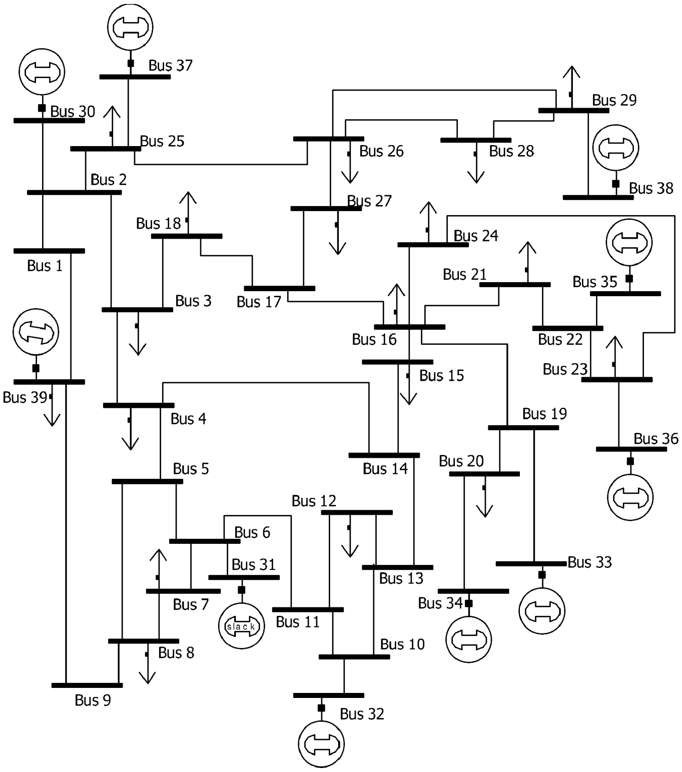


Figure 3.10: IEEE 39 bus power system.

In this experiment, to better evaluate our dynamic detector, our system does not implement SE.

We assume that the attackers can inject k measurements which are randomly chosen to generate Gaussian distributed attack vectors $a \sim \mathcal{N}(0, 0.5)$. We also test the scenario that the attacking vectors are derived from real measurement which will fail to be detected by most state-of-art detectors. In this experiment, the attackers try to inject a false generator trip event which is collected in advance, and we define attacking capability as $\frac{k}{n}$ where n is the total number of measurements. We evaluate the performance of our dynamic FDI attack detection framework on the classification results for the test set. We train a neural network with ten training epochs to minimize the loss function in Equation 3.16. For the experiment, we apply a 60% / 20% / 20% train / validation / test split, with a grid search to determine the best τ.

We illustrate the results of our anomaly detection system in Figure 3.11. From the figure, it is clear that our proposed detection mechanism can achieve the detection accuracy above

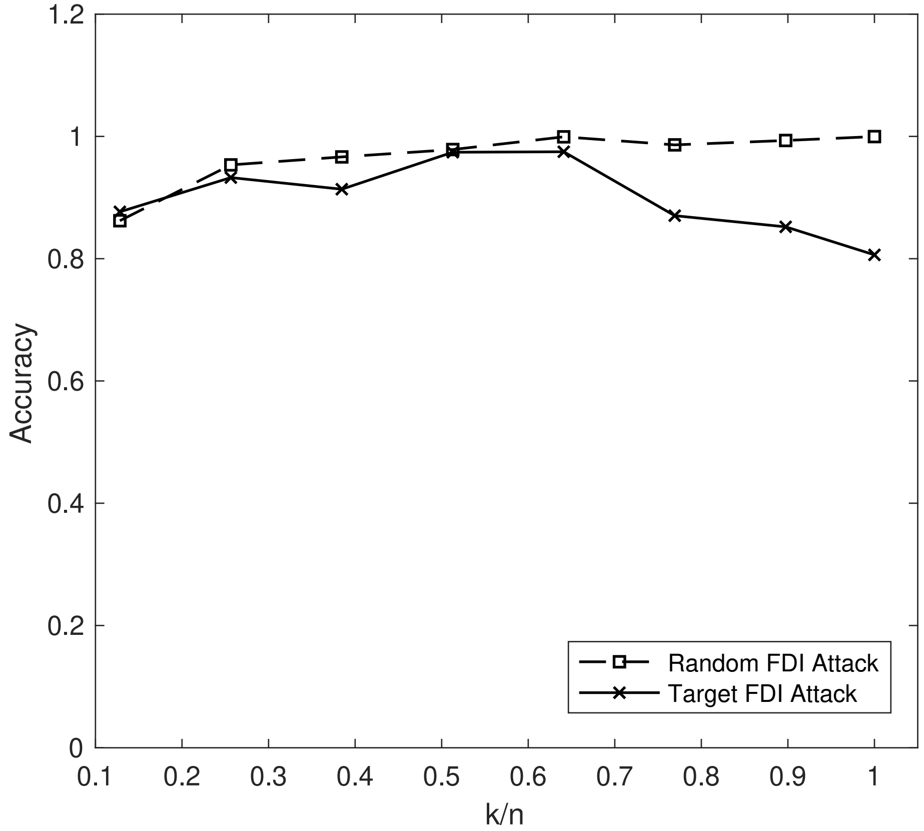


Figure 3.11: The accuracy of detecting FDI attacks for different the number of the compromised buses.

90% for random FDI attacks when $\frac{k}{n}$ is high. However, we also notice that our system has low accuracy when attacking power is low. In fact, this can be resolved by incorporating an SE detector (such as [95, 53]) which work well for limited attacking capability. In other words, our proposed two-level detection scheme is able to achieve high detection accuracy for different scenarios. For target FDI attacks, the injected data streams are carefully manipulated by a real event which is not considered for most SE bad data detection schemes. Our experiment validates that the dynamic features and network anomaly detector integration can support IDS for better performance. The simulation result in this case study also implies that the full in-depth knowledge of the power system is not required for the success of our dynamic detection scheme. Our system can be built at an early stage of an electricity network.

Chapter 4

Privacy-preserving Large Scale Security Constrained Optimal Power Flow

4.1 Introduction

In order to adequately supply the connected load while minimizing the operating costs, system operators need to solve the optimal power flow (OPF) problem subject to physical constraints and control limits of the power system [27]. However, due to the large-scale interconnected topology of transmission and distribution networks, the OPF model cannot ensure the demand-supply balance condition when the power system experiences unexpected failure and disconnection of components such as generators, transmission lines, transformers, etc., known as an outage or a contingency [64]. To address this issue, security requirements that ensuring the power system to continue its reliable operation during contingency scenarios need to be performed with the OPF problem, which is referred to as the security-constrained optimal power flow (SCOPF) problem in the literature [6, 83]. The optimal solution of the SCOPF problem produces the minimal cost generation dispatch while still assures that the power system remains balanced and no operational constraints are violated in both the normal state and contingencies [6, 83].

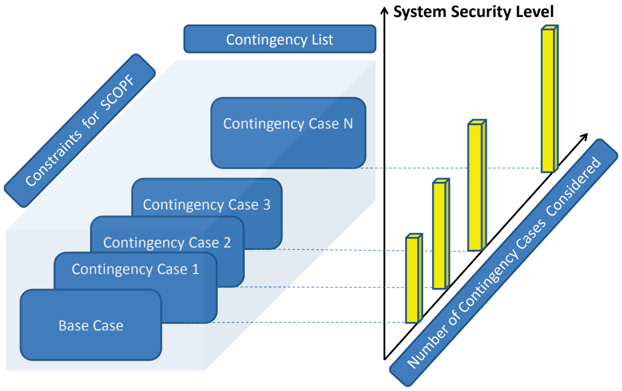


Figure 4.1: The system security level is improved by solving the SCOPF, which takes into account a number of contingency cases in a dedicate selected contingency list.

The contingency analysis is performed by independent system operators (ISOs) to ensure the reliable operation of power systems in both normal case and contingencies. By taking into account both pre-contingency constraints and post-contingency constraints in the SCOPF, the security of the power system can be significantly improved [25]. An illustration of the SCOPF is shown in Figure 4.1. The system security level is improved by taking into account a number of contingencies in a dedicate selected contingency list. The SCOPF is commonly classified into two major types: the preventive model [6] and corrective model [83]. The preventive SCOPF formulation seeks the minimum cost dispatch solution in the normal state that requires the normal-state variables to be feasible for all pre-specified contingency conditions, i.e., the control variables are not allowed to reschedule in contingency scenarios. However, this model makes the solution to be more conservative and may incur, in general, a higher operation cost [6].

The corrective SCOPF model, on the other hand, permits system operators to adjust post-contingency control variables such as power generation outputs and line power flows within a certain limit to eliminate any violation caused by the contingency. Due to the capability of adjusting control variables, the corrective SCOPF model often produces the optimal solution that has a lower total generation cost than the preventive model [99].

Despite the economic benefit of the corrective SCOPF model, its formulation generates additional variables, which sharply increase the problem size when numerous contingencies are taken into account. The large-scale formulated problem may result in excessive memory usage and unacceptable computation time [98]. Recently, cloud computing has demonstrated its massive potential for tremendously speeding up intensive computation while reducing the cost. Hence, outsourcing the SCOPF problem to the cloud has emerged as a promising solution to the challenges as mentioned above. Nonetheless, the fact that the operation takes place entirely at a third party will inevitably raise privacy concerns about data sensitivity. Sensitive power grid data can be captured by the cyber attacker and used to initiate a more sophisticated attack (e.g., false data injection attack [75]) which could have potentially catastrophic consequences. Alternatively, generic secure multi-party computation can implement any algorithm in principle, allowing the utility companies to take advantage of the aforementioned outsourcing paradigms while protecting the privacy of its operational data.

In this chapter, we present a decentralized structure of outsourcing paradigm and a distributed privacy-preserving algorithm to demonstrate the feasibility of solving the corrective SCOPF problem without losing data privacy. The basic idea of the proposed scheme is to let each substation encrypt their private data after which a third party performs the SCOPF algorithm over the encrypted data without decrypting it. The third party then sends the encrypted result to the ISO company, which can be decrypted using the pre-distributed secret key. This is accomplished by leveraging additive Homomorphic Encryption (such as the Paillier cryptosystem [97]). However, according to the additive homomorphism property, we cannot directly solve the SCOPF problem using any available methods. In this work, we leverage both alternating direction method of multipliers (ADMM) [21] and gradient projection algorithm [16] to transform the SCOPF problem into a solvable problem for the additive homomorphic cryptosystem.

Note that even though the proposed scheme is based on ADMM and gradient projection, it can also be easily extended to other sophisticated optimization algorithms. Also, our proposed method is not limited to solve the SCOPF problem; it can also be applied to any other optimization problems that involve ADMM or gradient projection such as Internet congestion control and power system state estimation.

4.1.1 Challenges

The OPF is of great importance to ISO power markets, and is solved recursively from every hour to every day. Thus, OPF has potentials to be targeted by attackers for sniffing attacks (See Section 2.6.2). What's more, a SCOPF problem contains high confidential information, such as local power constraints and bus connections. In this case, protecting SCOPF computation is particularly crucial. Traditional cryptography is widely utilized to secure the communication channels between different entities. Nonetheless, it is hard to figure out how to secure the computation process. For public key cryptosystem, variables must be decrypted in order correctly calculate the optimal flow, thus still suffering sniffing attacks. Taking the above requirements into consideration, we choose to use secure multi-party computation to secure the SCOPF computation which can perform calculations on encrypted data.

However, cryptography comes with a cost, especially for secure multi-party computation algorithms. Therefore, it is also necessary to figure out how to minimize the computation and communication overhead. Possible solutions include using high-performance computers or special designed cryptography hardware, and distributed computing. Taking cost into account, it is impractical for ISOs to employ high-performance computers or specially designed cryptography hardware. On the other hand, it is also not straightforward to develop a distributed multi-party secure computation algorithm for the SCOPF problem because of its large scale.

4.1.2 Outlines

By presenting privacy-preserving computation for the large-scale SCOPF problem the main contributions of this chapter can be summarized as follows:

- To our best knowledge, this work is the first to consider the privacy-preserving method for the SCOPF problem.

- Decomposed the SCOPF problem into independent subproblems using ADMM and gradient projection which are solved by Homomorphic Encryption.

- Both ADMM and gradient projection algorithms are proposed in a privacy-preserving manner.

We organize the rest of this chapter as follows. Section 4.2 reviews the related works. Section 4.3 introduces the architecture and threat model. Section 4.4 explains the components of privacy-preserving SCOPF. The section continues to presents the optimization techniques and summarizes our proposed privacy-preserving SCOPF scheme along with security analysis in Section 4.5. Section 4.6 presents the experiment results of our proposed scheme.

4.2 Related Work

Due to the large-scale nature of the SCOPF problem, recent research has tried to propose different methodologies to handle the SCOPF problem. The contingency filtering techniques have been developed in [8, 24] to discard contingencies that do not affect the optimal solution. An exact method to obtain the global optimal solution using a branch-and-bound algorithm has been proposed in [98]. The Benders decomposition techniques to handle each contingency separately and check the feasibility of the optimization problem has been applied in [123] to achieve computational efficiency. The works in [99, 74] apply the ADMM decomposition method to design a parallel computing framework, which can be executed simultaneously on multiple computers to reduce the running time for the algorithms.

Previously, researchers mainly focus on the privacy of customer data in the smart grid. Giaconi et al. [46] studied information leakage in a smart meter system, and the privacy can be partially preserved by a low-complexity policy which can approach the theoretical lower bound. This scheme only guarantees lower the information leakage rate in limited scenarios. Other methods for protection consumers' privacy can be found in [37, 11, 49]. Additionally, only a few researches have been published to address sensitive operational data for the power system. [122] tries to secure outsourcing of widely applicable linear programming (LP) computations by applying affine mapping on decision variables, which will transform the original vector space to a different one. [116] proposes a novel scheme that enables privacy preserving multi-party spectral estimations, which conduct spectral estimation directly over the encrypted synchrophasors to limit privacy breaches. However, both works only deal with the minimal scenario which cannot be applied to solve more complicated problems such as SCOPF.

Furthermore, privacy-preserving computations have received significant attention in areas other than the smart grid. A common approach makes use of two-party computation is based on Yao's garbled circuits. [89] shows the feasibility of designing a system that performs matrix factorization, a popular method used in a variety of modern recommendation systems. Several frameworks implement Yao's garbled circuits and describe many applications [66]. However, due to the inherent serial property, recent work [88] shows that the garbled circuit is 2^{14} times slower than computing in plaintext. To deal with the slowness of garbled circuit, some works introduced hybrid approaches, such as combining Homomorphic Encryption and garbled circuits for regression [89], face [106] and fingerprint recognition [39], and combining secret sharing with garbled circuits for learning a decision tree [4]. Nonetheless, the performance of those hybrid approaches is primarily affected by the network connection, which does not apply to our scheme.

4.3 System Architecture

4.3.1 System Model

Our system is designed for one or multiple entities who want to solve the large-scale SCOPF problem with a limited computational resource. The system model of the proposed scheme is captured in Figure 4.2 where the notation is given in Table 4.1. The proposed system model involves four different entities: Control Center (CC), Balancing Authority (BA), Server (S), and Cryptographic Provider (CP).

Control Center refers to as the operator of the regional power system who solves the SCOPF problem to minimize generation costs, market surplus, and losses, etc. An independent system operator (ISO) firstly prepares a SCOPF problem $\mathcal{F}$ which contains the objective function and constraints and send initiation request to each balancing authority. In our scheme, control center initials a SCOPF problem and sends it to the server.

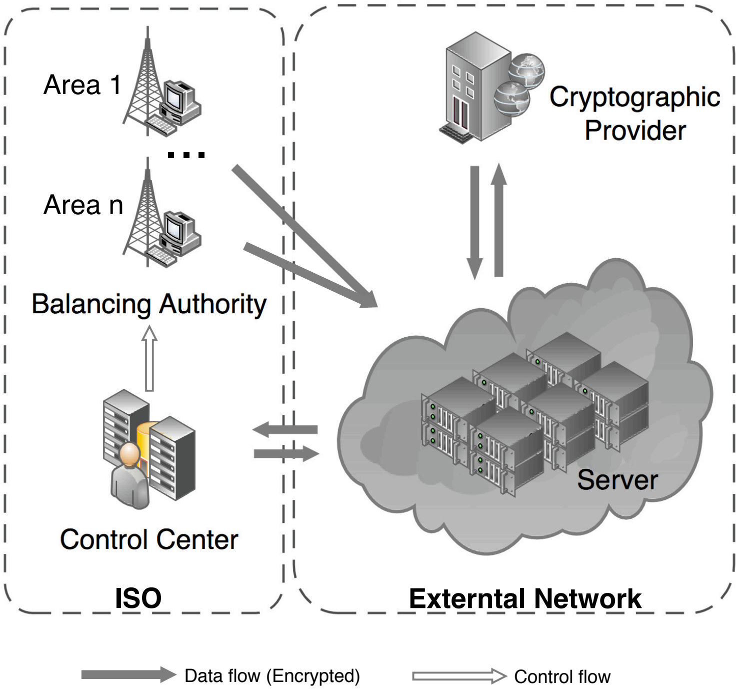


Figure 4.2: The architecture of proposed scheme

Balancing Authority is usually a substation in the power system $\mathcal{F}$ which contains local operation data and contingencies. In our scheme, after receiving a request command from the control center, balancing authorities generate encrypted local data and upload it to the server.

Server only stores the public key K_{pub}. Upon receiving the encrypted SCOPF problem $\mathcal{F}_c$ from the ISO, the server executes privacy-preserving algorithm over $\mathcal{F}_c$, and finally returns encrypted optimal $[\mathbf{P}^{g,c}]$ to the ISO. The contribution of our system is to ensure that the server learns nothing about the power flow while still being capable of computing the optimal state of the power system.

Cryptographic Provider is a third party that initializes the system by assigning setup parameters to each party and is needed for a short one-round online step in each iteration while the server computes the model.

Table 4.1: Notation definitions

Variable	**Description**
$\mathcal{G}$	The set of generators
$\mathcal{N}$	The set of buses
$\mathcal{B}$	The set of branches
$\boldsymbol{\theta}^c \in \mathbb{R}^{\lvert\mathcal{N}\rvert}$	The vector of voltage angles
$\mathbf{P}^{g,c} \in \mathbb{R}^{\lvert\mathcal{G}\rvert}$	The vector of real power flows
f_i^g	The generation cost function
$\mathbf{P}_i^{g,0}$	The displaceable real power of each individual generation unit i for pre-contingency configuration
$\mathbf{B}^c_{bus} \in \mathbb{R}^{\lvert\mathcal{N}\rvert\times\lvert\mathcal{N}\rvert}$	The power network system admittance matrix
$\mathbf{B}^c_f \in \mathbb{R}^{\lvert\mathcal{B}\rvert\times\lvert\mathcal{N}\rvert}$	The branch admittance matrix
$\mathbf{P}^{d,c} \in \mathbb{R}^{\lvert\mathcal{N}\rvert}$	The real power demand
$\mathbf{C}^{g,c} \in \mathbb{R}^{\lvert\mathcal{N}\rvert\times\lvert\mathcal{G}\rvert}$	The sparse generator connection matrix, whose element (i,j) element is 1 if generator j is located at bus i and 0 otherwise
$\mathbf{F}_{max}$	The vector for the maximum power flow
$\overline{\mathbf{P}^{g,c}}$	The upper bound of real power generation
$\underline{\mathbf{P}^{g,c}}$	The lower bound of real power generation
$\boldsymbol{\Delta}_c$	The pre-defined maximal allowed variation of power outputs
K_{priv}	The private key of a cryptosystem
K_{pub}	The public key of a cryptosystem
$\mathcal{E}$	A Paillier cryptosystem
$\mathcal{F}$	The plaintext of a SCOPF problem, denoted as a collection of base case and all the contingencies
$\mathcal{F}_c$	The encrypted SCOPF problem stored in the server
$[x]$	The encryption form of x using Paillier cryptosystem

4.3.2 Design Goals and Threat Model

To enable secure, efficient, and accurate solving SCOPF over the encrypted problem under the above model. Our goal is to ensure the security of our algorithms using the secure two-party computation framework for semi-honest adversaries (or honest-but-curious adversaries) [47]. The specific requirements are summarized as follows:

- *Data Privacy:* The server and cryptographic provider could not reveal the power flow using the statistical information during the computation process.

- *System Confidentiality:* Given an encrypted SCOPF problem $\mathcal{F}_c$, The server and cryptographic provider are not able to recover key information of the power grid, such as power demand, bus limits, and generation load.

- *Efficiency:* The scheme aims to achieve efficiency by offloading the computation to the server and by using the parallel approach.

- *Accuracy:* The difference between the result that calculated by the proposed parallel scheme and traditional centralized method will not exceed the threshold e.

In our system, we assume that the Server is able to produce a correct model. Thus, we do not concern with a malicious server who tries to disrupt the optimization algorithm to output an incorrect result. However, the server is motivated to learn information about private data stored on the server since this data can potentially be sold to other parties, e.g., black market. In our scenario, consider the server is compromised by a semi-honest adversary. The adversary aims to learn the SCOPF problem and the optimization result as much as possible by analyzing all the input and output of this party. That is to say, the server can conduct a ciphertext-only attack (COA) [47] in this model. However, the adversary cannot prevent this party from executing the algorithm faithfully. Hence, we do not consider an adversary who will intentionally corrupt the operation to generate misleading results.

4.4 Preliminaries

To better understand our scheme, in this section, we review SCOPF and necessary components of the proposed algorithm.

4.4.1 SCOPF Problem Formulation

The corrective SCOPF problem finds the optimal dispatch solution for the power network while satisfying security criteria in which the system operator is allowed to re-adjust control variables after a contingency occurs. This capability gives the system operator a time window to adjust control variables in order to eliminate any violations caused by the contingency. The general formulation of the corrective SCOPF problem can be compactly formulated as follows [83]:

$$\begin{aligned}
\min_{\mathbf{x}^0,\ldots,\mathbf{x}^C;\mathbf{u}^0,\ldots,\mathbf{u}^C} \quad & f^0(\mathbf{x}^0,\mathbf{u}^0) && (4.1)\\
\text{subject to} \quad & \mathbf{g}^0(\mathbf{x}^0,\mathbf{u}^0) = 0, && (4.2)\\
& \mathbf{h}^0(\mathbf{x}^0,\mathbf{u}^0) \leq 0, && (4.3)\\
& \mathbf{g}^c(\mathbf{x}^c,\mathbf{u}^c) = 0, \quad \forall c \in \mathcal{C}, && (4.4)\\
& \mathbf{h}^c(\mathbf{x}^c,\mathbf{u}^c) \leq 0, \quad \forall c \in \mathcal{C}, && (4.5)\\
& |\mathbf{u}^0 - \mathbf{u}^c| \leq \mathbf{\Delta}_c, \quad \forall c \in \mathcal{C}, && (4.6)
\end{aligned}$$

where $\mathcal{C} = \{1, 2, \ldots, C\}$ is the set of postulated contingencies, superscript c denotes variables and constraints associated with the c-th contingency, superscript 0 represents the base case (pre-contingency state), $\mathbf{x}$ and $\mathbf{u}$ denote state variables and control variable. The constraints in (4.2) and (4.3) denote the set of equality and inequality constraints associated with the operation of power system such as transmission line limits, power flow equations, etc., in the base case. Similarly, (4.4) and (4.5) represent the operational constraints of power system when switching into contingency states. The last constraints in (4.6) are the coupling constraints between the base case and post-contingency, which mean that the deviation of control variables between the normal state and contingency states must be within the allowable adjustment limit, denoted by $\mathbf{\Delta}_c$.

Based on the standard form of the SCOPF problem, there are some variations on the objective function and constraints for the SCOPF problem in alternating current (AC) and direct current (DC) power networks. For the sake of computational tractability of the proposed privacy scheme, we consider a DC power network with the objective function is to minimize the total generation cost while ensure the security requirements for the power system. Therefore, the corrective SCOPF problem can be simplified as follows [74]:

$$\begin{aligned}
\min_{\boldsymbol{\theta}^0,\ldots,\boldsymbol{\theta}^C;\mathbf{P}^{g,0},\ldots,\mathbf{P}^{g,C}} \quad & \sum_{i\in\mathcal{G}} f_i^g(\mathbf{P}_i^{g,0}) && (4.7)\\
\text{subject to} \quad & \mathbf{B}_{bus}^0\boldsymbol{\theta}^0 + \mathbf{P}^{d,0} - \mathbf{C}^{g,0}\mathbf{P}^{g,0} = 0, && (4.8)\\
& \mathbf{B}_{bus}^c\boldsymbol{\theta}^c + \mathbf{P}^{d,c} - \mathbf{C}^{g,c}\mathbf{P}^{g,c} = 0, && (4.9)\\
& |\mathbf{B}_f^0\boldsymbol{\theta}^0| - \mathbf{F}_{max} \leq 0, && (4.10)\\
& |\mathbf{B}_f^c\boldsymbol{\theta}^c| - \mathbf{F}_{max} \leq 0, && (4.11)\\
& \underline{\mathbf{P}^{g,0}} \leq \mathbf{P}^{g,0} \leq \overline{\mathbf{P}^{g,0}}, && (4.12)\\
& \underline{\mathbf{P}^{g,c}} \leq \mathbf{P}^{g,c} \leq \overline{\mathbf{P}^{g,c}}, && (4.13)\\
& |\mathbf{P}^{g,0} - \mathbf{P}^{g,c}| \leq \mathbf{\Delta}_c, && (4.14)\\
& i \in \mathcal{G}, \quad c = 1,\ldots,C, && (4.15)
\end{aligned}$$

where $\mathbf{P}_i^{g,0}$ is the generation output of each individual generator for pre-contingency configuration, $f_i^g(\mathbf{P}_i^{g,0})$ represents the generation cost function of the generator using the following function

$$f_i^g(\mathbf{P}_i^{g,0}) = a_i(\mathbf{P}_i^{g,0})^2 + b_i\mathbf{P}_i^{g,0} + c_i, \qquad (4.16)$$

where a_i, b_i and c_i are the cost coefficients. The constraints (4.8) and (4.9) are the nodal load-flow equations. The inequality constraints (4.10) and (4.11) are the transmission line limits. Constraints (4.12) and (4.13) are the power generation limits. Constraints (4.14) enforce the maximum adjusting limits for the generation units when switching into post-contingency states. The key notations of the above problem can be found in Table 4.1.

The problem in (4.7)-(4.15) is convex and can be solved by a central controller using convex optimization techniques [22]. However, due to numerous contingencies are incorporated into the model, the formulated problem becomes a large-scale optimization problem, which makes the centralized computational framework impractical. We will propose a parallel computation algorithm using the ADMM decomposition technique in the next section.

4.4.2 Paillier Cryptosystem

Paillier cryptosystem is a public key based additive homomorphic cryptosystem first proposed in [97] and further generalized by Damgård and Jurik [33]. Here, additive homomorphic cryptosystem means it can compute the sum of two values in the encrypted domain, which means given the encryption of a and b, we can get encryption $a + b$ without decryption. In this case, the Paillier cryptosystem is very useful for privacy-preserving applications. In this subsection, we will illustrate the key components of the Paillier cryptosystem.

Key Generation

First, two big prime numbers p and q are selected to compute $n = pq$ and $\lambda = \text{lcm}(p-1, q-1)$. Then choose a random integer g where $g \in \mathbb{Z}^*_{n^2}$ ($\mathbb{Z}^*_{n^2} = \{z | z \in \mathbb{Z}, gcd(z, n^2) = 1\}$ and $\mathbb{Z}$ denotes the set of all integers). The private key is λ and the public key is the tuple (n, g).

Encryption

To encrypt a value m where $m < n$ in the typical setting, we need to select a random value $r \in \mathbb{Z}^*_n$. Then the ciphertext can be computed as: $c = g^m \cdot r^n \bmod n^2$. Note that in our scenario, m is not necessarily positive. Therefore, we divide the encryption space in two parts and then m is in the range of $(-n/2, n/2)$.

Decryption

To decrypt a ciphertext c, the plaintext message can be computed as:

$$m = D(c) = \frac{L(c^{\lambda} \bmod n^2)}{L(g^{\lambda} \bmod n^2)} \bmod n, \tag{4.17}$$

where function L is defined as $L(u) = \frac{u-1}{n}$.

Additive Homomorphism

Given the ciphertext of m_1 and m_2, the product of two ciphertexts will decrypt to the sum of their corresponding plaintexts:

$$D(E(m_1, r_1) \cdot E(m_2, r_2) \bmod n^2) = m_1 + m_2 \bmod n. \tag{4.18}$$

Moreover, given a ciphertext and a plaintext, we can compute the sum of the corresponding plaintexts:

$$D(E(m_1, r_1) \cdot g^{m_2} \bmod n^2) = m_1 + m_2 \bmod n. \tag{4.19}$$

Homomorphic Multiplication of Plaintext

By raising the ciphertext to a constant k, we can get the encryption of the product of the plaintext and the constant:

$$D(E(m_1, r_1)^k \bmod n^2) = km_1 \bmod n. \tag{4.20}$$

However, given the encryptions of two plaintexts, there is no direct way to compute the product of these messages without knowing the private key.

4.4.3 The ADMM Method

The ADMM is a powerful algorithm that is proposed to solve convex optimization. Its general idea is to solve small local subproblems, which are coordinated to find a solution to a global problem by blending the benefits of dual decomposition and augmented Lagrangian methods for constrained optimization. The general form of ADMM is described as follows:

$$\min_{\mathbf{x},\mathbf{z}} \quad f(\mathbf{x}) + g(\mathbf{z}) \tag{4.21}$$

$$\text{subject to} \quad \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z} = \mathbf{c}, \tag{4.22}$$

where $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{z} \in \mathbb{R}^m$, $\mathbf{A} \in \mathbb{R}^{p\times n}$, $\mathbf{B} \in \mathbb{R}^{p\times m}$ and $\mathbf{c} \in \mathbb{R}^p$. Functions f and g are closed, convex and proper. The scaled augmented Lagrangian can be expressed as:

$$\mathcal{L}_\rho(\mathbf{x}, \mathbf{z}, \boldsymbol{\mu}) = f(\mathbf{x}) + g(\mathbf{z}) + \frac{\rho}{2}\|\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z} - \mathbf{c} + \boldsymbol{\mu}\|_2^2, \tag{4.23}$$

where $\rho > 0$ is the penalty parameter and $\boldsymbol{\mu}$ is the scaled dual variable. Using the scaled dual variable, $\mathbf{x}$ and $\mathbf{z}$ can be updated in a Guass-Seidel fashion. At each iteration k, the update process can be expressed as:

$$\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} f(\mathbf{x}) + \frac{\rho}{2}\|\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z}^k - \mathbf{c} + \boldsymbol{\mu}^k\|_2^2, \tag{4.24}$$

$$\mathbf{z}^{k+1} = \arg\min_{\mathbf{z}} g(\mathbf{z}) + \frac{\rho}{2}\|\mathbf{A}\mathbf{x}^{k+1} + \mathbf{B}\mathbf{z} - \mathbf{c} + \boldsymbol{\mu}^k\|_2^2. \tag{4.25}$$

Finally, the scale dual variable is updated by:

$$\boldsymbol{\mu}^{k+1} = \boldsymbol{\mu}^k + \mathbf{A}\mathbf{x}^{k+1} + \mathbf{B}\mathbf{z}^{k+1} - \mathbf{c}. \tag{4.26}$$

4.5 Privacy-preserving SCOPF

The basic idea of this algorithm is to use Paillier cryptosystem to solve multiplication, addition, and subtraction without leaking any information about the input. However, the objective function of the SCOPF problem is not necessarily linear and usually in the quadratic form. To address this challenge, we will reformulate the SCOPF problem using the ADMM and gradient projection algorithms in this section.

4.5.1 Reformulating the Problem

The SCOPF problem in (4.7)-(4.15) contains a large number of constraints. However, we can separate constraints (4.8)-(4.13) into the normal state and each contingency state. The only constraints in (4.14) are coupling between the normal state and contingency state. In order to make constraints in (4.14) to be separable, we define auxiliary variables

$$\mathbf{P}^{g,c} = \mathbf{P}_o^{g,c}, \forall c \in \mathcal{C}, \tag{4.27}$$

where each auxiliary variable $\mathbf{P}_o^{g,c}$ can be interpreted as local copy of $\mathbf{P}^{g,c}$ at the normal state. Then, constraint (4.14) now can be rewritten separately for the normal state and contingencies as

$$|\mathbf{P}^{g,0} - \mathbf{P}_o^{g,c}| \leq \boldsymbol{\Delta}_c. \tag{4.28}$$

The constraints (4.8), (4.10), (4.12), and (4.28) are now consisting of variables in the normal state only, while constraints (4.9), (4.11), and (4.13) contains variables in contingencies. To facilitate the presentation, we define the feasible sets in the normal state, $\mathcal{F}^0$, and each contingency state, $\mathcal{F}^c$, as

$$\begin{aligned} \mathcal{F}^0 &= \{(\mathbf{P}^{g,0}, \mathbf{P}_o^{g,c}) | (4.8), (4.10), (4.12), (4.28)\}, \\ \mathcal{F}^c &= \{(\mathbf{P}^{g,c}) | (4.9), (4.11), (4.13)\}, \forall c \in \mathcal{C}. \end{aligned}$$

Then, the problem in (4.7)-(4.15) can be rewritten as

$$\min_{\boldsymbol{\theta}^0,\ldots,\boldsymbol{\theta}^C;\mathbf{P}^{g,0},\ldots,\mathbf{P}^{g,C}} \quad \sum_{i \in \mathcal{G}} f_i^g(\mathbf{P}_i^{g,0}) \tag{4.29}$$

$$\text{subject to} \quad \{\mathbf{P}^{g,0}, \mathbf{P}_o^{g,c}\} \in \mathcal{F}^0, \tag{4.30}$$

$$\{\mathbf{P}^{g,c}\} \in \mathcal{F}^c, \forall c \in \mathcal{C}, \tag{4.31}$$

$$\mathbf{P}^{g,c} = \mathbf{P}_o^{g,c}, \forall c \in \mathcal{C}. \tag{4.32}$$

Algorithm 1 Distributed SCOPF

```
Input: $B_{bus}^c, B_f^c, A^{g,c}, \mathbf{P}^{d,c}, \overline{\mathbf{P}^{g,c}}, \underline{\mathbf{P}^{g,c}}, \mathbf{\Delta}_c$
Initialize: $\theta^c, \mathbf{P}^{g,c}, \mathbf{p}^c, \boldsymbol{\mu}^c, \rho^c, k = 0$
while not converge do
    $\mathbf{P}^{g,0}$ - update
    $\mathbf{P}^{g,0^{k+1}} = \arg\min_{\mathbf{P}^{g,0},\mathbf{P}} \sum_{i\in\mathcal{G}} f_i^g(\mathbf{P}_i^{g,0}) - \sum_{c=1}^{C} \boldsymbol{\mu}^c \mathbf{P}_o^{g,c} + \frac{\rho}{2} \sum_{c=1}^{C} \|\mathbf{P}^{g,c} - \mathbf{P}_o^{g,c}\|_2^2$
    s.t.  $\{\mathbf{P}^{g,0}, \mathbf{P}_o^{g,c}\} \in \mathcal{F}^0$

    $P^{g,c}$-update, distributively at each node:
    $\mathbf{P}^{g,c^{k+1}} = \arg\min_{\mathbf{P}^{g,c}} \boldsymbol{\mu}^c \mathbf{P}^{g,c} + \frac{\rho}{2} \|\mathbf{P}^{g,c} - \mathbf{P}_o^{g,c}\|_2^2$
    s.t.  $\{\mathbf{P}^{g,c}\} \in \mathcal{F}^c$.
    $\boldsymbol{\mu}^{c^{k+1}} = \boldsymbol{\mu}^{c^k} + \rho(\mathbf{P}^{g,c} - \mathbf{P}_o^{g,c})$
    Adjust penalty parameter $\rho^c$ when necessary
    $k = k + 1$
end while
return $\theta^c, \mathbf{P}^{g,c}, c = 0, ..., C$
```

The augmented Lagrangian can then be calculated as:

$$\begin{aligned}
&\mathcal{L}_\rho(\mathbf{P}^{g,0}, ..., \mathbf{P}^{g,C}; \mathbf{P}_o^{g,1}, ..., \mathbf{P}_o^{g,C}; \boldsymbol{\mu}^1, ..., \boldsymbol{\mu}^C) \\
&= \sum_{i\in\mathcal{G}} f_i^g(\mathbf{P}_i^{g,0}) + \sum_{c=1}^{C} \boldsymbol{\mu}^c(\mathbf{P}^{g,c} - \mathbf{P}_o^{g,c}) \\
&\quad + \frac{\rho}{2} \sum_{c=1}^{C} \|\mathbf{P}^{g,c} - \mathbf{P}_o^{g,c}\|_2^2.
\end{aligned} \tag{4.33}$$

Based on the Lagrangian function, we can decompose the problem in (4.7)-(4.15) into $C + 1$ subproblems as follows:

$\mathbf{P}^{g,0}$-update

At k^{th} iteration, the $\mathbf{P}^{g,0}$-update solves the base scenario with square regularization terms enforce by the coupling constraints and expressed as:

$$\begin{aligned}
\mathbf{P}^{g,0}[k+1] &= \arg\min \mathcal{L}_\rho(P^{g,0}) \\
&= \underset{\mathbf{P}^{g,0},\mathbf{p}}{\arg\min} \sum_{i\in\mathcal{G}} f_i^g(\mathbf{P}_i^{g,0}) - \sum_{c=1}^{C} \boldsymbol{\mu}^c \mathbf{P}_o^{g,c} \\
&+ \frac{\rho}{2}\sum_{c=1}^{C} \|\mathbf{P}^{g,c} - \mathbf{P}_o^{g,c}\|_2^2 \qquad (4.34)\\
\text{s.t.} \quad & \{\mathbf{P}^{g,0}, \mathbf{P}_o^{g,c}\} \in \mathcal{F}^0. \qquad (4.35)
\end{aligned}$$

$\mathbf{P}^{g,c}$-update

The remaining C subproblems are associated with variables in contingency scenarios. Each contingency can be solved in parallel at different computing nodes as

$$\begin{aligned}
\mathbf{P}^{g,c}[k+1] &= \arg\min \mathcal{L}_\rho(P^{g,c}) \\
&= \underset{\mathbf{P}^{g,c}}{\arg\min}\, \boldsymbol{\mu}^c \mathbf{P}^{g,c} + \frac{\rho}{2}\|\mathbf{P}^{g,c} - \mathbf{P}_o^{g,c}\|_2^2 \\
\text{s.t.} \quad & \{\mathbf{P}^{g,c}\} \in \mathcal{F}^c. \qquad (4.36)
\end{aligned}$$

$\boldsymbol{\mu}$ -update

The computation of updating dual variables are also linear and can be performed locally at c^{th} computing utility as:

$$\boldsymbol{\mu}^c[k+1] = \boldsymbol{\mu}^c[k] + \rho(\mathbf{P}^{g,c} - \mathbf{P}_o^{g,c}). \qquad (4.37)$$

4.5.2 Solving Subproblems

In this section, we explain how to solve each subproblem using Homomorphic Encryption in a simple way. We formulate each subproblem as a quadratic optimization problem:

$$\begin{aligned} \arg\min_{x} \quad & x^T \cdot A \cdot x + b^T \cdot x && (4.38)\\ \text{subject to} \quad & B \cdot x = d, && (4.39)\\ & u \leq C \cdot x \leq v, && (4.40)\\ & s \leq x \leq t. && (4.41) \end{aligned}$$

The linear inequality constraints are difficult to deal by basic operations in their current form. So we further simplify this problem by introducing a dummy variable $y = C \cdot x$. Then, we obtain the problem:

$$\begin{aligned} \arg\min_{x} \quad & x^T \cdot A \cdot x + b^T \cdot x && (4.42)\\ \text{subject to} \quad & B \cdot x = d, && (4.43)\\ & C \cdot x - y = 0, && (4.44)\\ & u \leq y \leq v, && (4.45)\\ & s \leq x \leq t. && (4.46) \end{aligned}$$

This subproblem itself can be solved by ADMM. We denote the augmented Lagrangian:

$$\begin{aligned} \mathcal{L}_\beta(x, y; w_1, w_2) :=& x^T \cdot A \cdot x + b^T \cdot x + \frac{\beta}{2}\|Bx - d - w_1\|_2^2 \\ &+ \frac{\beta}{2}\|Cx - y - w_2\|_2^2, \qquad (4.47) \end{aligned}$$

where w_1 and w_2 are scaled Lagrange multipliers, and β is a penalty parameter. The iteration of ADMM is

$$x^{k+1} = \arg\min_{x} \mathcal{L}_\beta(x, y^k; w_1^k, w_2^k) \quad \text{subject to} \quad s \leq x \leq t; \tag{4.48}$$

$$y^{k+1} = \arg\min_{y} \mathcal{L}_\beta(x^{k+1}, y; w_1^k, w_2^k) \quad \text{subject to} \quad u \leq y \leq v; \tag{4.49}$$

$$w_1^{k+1} = w_1 k - (Bx^{k+1} - d); \tag{4.50}$$

$$w_2^{k+1} = w_2^k - (Cx^{k+1} - y^{k+1}). \tag{4.51}$$

The x-subproblem does not have a closed-form solution. It is solved by another iteration of gradient projection: $x \leftarrow proj_{[s;t]}(x - \alpha\nabla_x \mathcal{L}_\beta(x, y; w_1, w_2)$. Since $\mathcal{L}$ is quadratic, $(x - \alpha\nabla_x \mathcal{L}_\beta(x, y; w_1, w_2)$ is linear and reduces to matrix-vector multiplications and vector-vector sums/differences.

To summarize, the numerical operations of the above algorithms include and only include:

- matrix-vector multiplications;
- vector-vector addition and subtraction;
- component-wise min and max.

4.5.3 Gradient Projection Algorithm

In the previous section, we reformulate the SCOPF problem to a set of subproblems which are solvable by additive Homomorphic Encryption. However, at the end of each subproblem, $\mathbf{P}^{g,0}$ and $\mathbf{P}^{g,c}$ are updated using the gradient projection algorithm. The gradient projection algorithm solves the bound constrained optimization problems by projecting the result of the gradient descent to the feasible set:

$$x_{k+1} = proj_{[s,t]}[x_k - \alpha\nabla f(x_k)]. \tag{4.52}$$

For the bound-constrained problems, the projection can be easily computed by setting:

$$proj_{[s,t]}(x) = mid(x, s, t), \tag{4.53}$$

where $mid\ldots$ is the median element of a set which cannot be directly solved by additive Homomorphic Encryption.

We now describe our privacy-preserving gradient projection algorithm. One challenging part is how to perform the comparison in the ciphertext domain. There are a few approaches to perform comparison efficiently. One naive way is sending the projection operation back to the ISO and calculate offline. Although this method is very efficient, it requires the ISO to keep online. Two methods that do not need the ISO to stay online are using specialized homomorphic encryption [119], or using garbled circuits [14]. Based on [20], the former is more efficient for comparison of encrypted values, the second is more efficient for comparison of unencrypted values. In our system, the lower and upper boundaries usually are power generation or bus limit which are sensitive information for the power grid. In this case, we are using homomorphic encryption to solve comparison.

The idea is to exploit the homomorphic property to obscure the inputs with an additive mask. Here $r_l, r_h \leftarrow (-2^l, 2^l)$ are in the message space of homomorphic encryption $\mathcal{E}$ and they follows that: $[x]^{[(r_l;r_h)]} = [(x \cdot r_l; x \cdot r_h)]$. To evaluate (4.52) over encrypted x, we need first calculate $[x_k] - \alpha \cdot [\nabla \mathcal{L}_x(x)]$ using its homomorphic property. Then, the server chooses a random mask r_l, r_h, obscures the difference $[x - s; x - t]$ as above, and sends the masked value to the Cryptographic Provider. The Cryptographic Provider can apply its decryption key and determine if a number is positive or negative. $proj_{[s,t]}(x)$ is hence solved by the cloud by simply checking the value of $(r_l; r_h)$ the result sent from the Cryptographic Provider.

The privacy-preserving gradient projection algorithm details in Algorithm 2. Here, we use $[x]$ to denote the encryption of x under the Paillier cryptosystem. Note that the computations in this algorithm are in the ciphertext domain and $[a] \cdot [b]^{-1} \bmod N^2$ in the ciphertext domain is equal to $a - b$ in the plaintext domain. Detailed secure proof of Algorithm 2 is in Section 4.5.7.

4.5.4 Dealing with Floating Point Numbers

Given its Homomorphic property, the Paillier cryptosystem has been tested in many scenarios. However, one limitation of the Paillier cryptosystem is that it can only work with integers. Although we can test our scheme only on an integer model, the SCOPF

Algorithm 2 Gradient Projection Algorithm

```
**Input:** S: data $[x_k]$, lower boundary $[s]$, upper boundary $[t]$.
**Input:** CP: private key $K_{priv}$
**Output:** $[x_{k+1}]$
S: $[x_k] \leftarrow [x_k] - \alpha \cdot [\nabla \mathcal{L}_x(x)]$
S: $[d_l] \leftarrow [x_k] \cdot [s]^{-1} \bmod n^2$
S: $[d_h] \leftarrow [x_k] \cdot [t]^{-1} \bmod n^2$
S choose two random integers $r_l, r_h \leftarrow (-2^l, 2^l)$
S: $[z_l] \leftarrow [d_l]^r_l \bmod n^2$
S: $[z_h] \leftarrow [d_h]^r_h \bmod n^2$
S sends $[z_l]$ and $[z_h]$ to B
CP decrypts $[z_l]$ and $[z_h]$
CP checks $z_l < 0$ and $z_h > 0$ and encrypts them using $K_{pub}$
CP sends S $[z_l < 0]$ and $[z_h > 0]$
**if** $z_l > 0 \oplus (r_l < 0)$ **then**
    $[x_{k+1}] = [s]$
**else if** $z_h < 0 \oplus (r_h > 0)$ **then**
    $[x_{k+1}] = [t]$
**else**
    $[x_{k+1}] = [x_k]$
**end if**
```

problem usually uses floating point numbers in the real world. Hence when we evaluate our system, we must adapt to it accordingly.

In practice, as discussed in Section 4.4.2, the Paillier cryptosystem involves only additions and multiplications. Therefore, in [116], the authors use a solution by multiplying each floating number with a constant K. However, the selection of K must guarantee that we do not overflow the Paillier cryptosystem plaintext space during the entire operation. To guarantee this, the selection of K must satisfy:

$$0 < K < \sqrt{\frac{2^{1024}}{2 \cdot m_{max}}}, \tag{4.54}$$

where m_{max} is the largest plaintext number in the system. The square root is due to the fact that for multiplications, the Paillier cryptosystem requires integers for both sides of the operation. Fortunately, due to constraints (4.10)-(4.13) introduced in the SCOPF problem, the range of the encrypted values is small enough to ensure that the selection of K can maintain high accuracy of the scheme.

Recall that, in our scheme, the global minima is found by iterations $P^{g,c} \leftarrow P^{g,c} - \alpha \cdot \nabla\mathcal{L}(P^{g,c})$ and at each iteration, the magnitude of x will increase by K. This is because both α and $\nabla\mathcal{L}(P^{g,c})$ are scaled to integers by multiplying K. Moreover, as mentioned in Section 4.4.2, the Paillier cryptosystem can only deal with multiplication directly, not division. As a consequence, the plaintext space will be overflowed after several rounds of iterations. The simplest way to rescale $P^{g,c}$ is to send it to an authorized party who has the secret key to decrypt it and sends the rescaled value back to the cloud. However, in reality, it is costly to find a trusted third party for this job since this party needs to keep online during the whole process. Another practical solution is to use the garbled circuit for operations that are not able to be solved by Homomorphic Encryption. In our case, the garbled circuit is considered computationally complex compared with Homomorphic Encryption if we only use the garbled circuit to solve divisions.

In [118], an efficient way to compute $[x \div d]$ from $[x]$ and d is proposed by additive blinding. Because B is not allowed to learn value x, it is additively blinded by a random number r which is the statistical security parameter. It leads to Algorithm 3, where the random number r is chosen as large as possible to ensure the best statistical hiding of x.

Algorithm 3 Approximate Division Algorithm

```
Input: S: $[x]$ and $d$.
Input: CP: $d$ and $K_{priv}$
Output: $[x \div d]$
S chooses a random number $r$ of size $\log_2 n - 1$
S computes $[z] = [x + r] = [x] \cdot [r] \bmod n$ and sends $[z]$ to CP.
CP decrypts $[z]$, and computes $z \bmod d$.
CP computes $c = z \div d$, encrypts it, and sends $[c]$ to S.
S computes $[x \div d] = [(z \div d) - (r \div d)] = [c] \cdot [r \div d]^{-1} \bmod n$.
```

4.5.5 Privacy-preserving SCOPF

Our Privacy Preserving SCOPF algorithm can fall into 4 phases:

- **Phase 1: Key Generation and Distribution.** After the system is activated, Cryptographic Provider initializes the Paillier cryptosystem as described in Section 4.4.2 with a public key K_{pub} and a private key K_{priv}. The Cryptographic Provider sends the private key to the control center and keeps a copy to itself and assigns the public key to server S and balancing authorities BA.

- **Phase 2: System Initialization.** With the public key generated by the Cryptographic Provider, each area can encrypt its subproblem and upload it to the server. The server then collects all the encrypted data and sets up the system by initializing the following variables: $[\theta^c], [\mathbf{P}^{g,c}], [\mathbf{P}_o^{g,c}], [\boldsymbol{\mu}^c], \rho, k = 0$. It should notice that ADMM converges to the optimum geometrically for the convex optimization problem [35], and the convergence time will be significantly reduced by using the warm start technique [92].

- **Phase 3: Privacy-preserving SCOPF.** With the encrypted input, the server securely operates the privacy-preserving SCOPF Algorithm 4, which calls two supporting algorithms, i.e., Algorithms 2 and 3. Since all computations are carried out in the ciphertext domain, no information about the measurement is revealed.

- **Phase 4: Result Decryption.** The ISO receives the encrypted $\theta^c, \mathbf{P}^{g,c}, c = 0, ..., C$ as a result of Algorithm 4 from the server. It then decrypts for the optimized variables using the private key. Also, since the uploaded power variables are multiplied by K, the optimized power variables should be divided by K accordingly.

4.5.6 Discussion

The proposed algorithm has several strengths that make them efficient and practical in real-world scenarios:

- First, both balancing authorities and control center do not need to stay online during the process. The ISO can leave the system after submitting the SCOPF problem and wait until final optimization is reached.

- Second, each local area can upload data in the encrypted form directly to the server to avoid the communication delay that costs by routing through the control center. The data integrity can be preserved using the Internet layer security protocol such as IPsec.

Algorithm 4 Privacy preserving SCOPF

```
Input: ISO: $[P^{g,0}], [\overline{\mathbf{P}^{g,c}}], [\underline{\mathbf{P}^{g,c}}], [\mathbf{P}^{d,c}], B^c_{bus}, B^c_f, A^{g,c}$,
Initialize: S: $[\theta^c], [\mathbf{P}^{g,c}], [\mathbf{P}^{g,c}_o], [\boldsymbol{\mu}^c], \rho, k = 0$
  while not converge do
    $P^{g,0}$-update:
    S computes $\nabla_{\mathbf{P}^{g,0}}\mathcal{L}_\rho(\mathbf{P}^{g,0^k})$
    S computes $\mathbf{P}^{g,0^{k+1}} \leftarrow proj_{[s,t]}(\mathbf{P}^{g,0^k} - \alpha\nabla_{\mathbf{P}^{g,0}}\mathcal{L}_\rho(\mathbf{P}^{g,0^k}))$
    /* Using Algorithm 2 */
    S computes $\mathbf{P}^{g,0^{k+1}}/K$
    /* Using Algorithm 3 */
    $\mathbf{P}^{g,c}$-update:
    while $c < C$ do
      S computes $\nabla_{\mathbf{P}^{g,c}}\mathcal{L}_\rho(\mathbf{P}^{g,c^k})$
      S computes $\mathbf{P}^{g,c^{k+1}} \leftarrow proj_{[u,v]}(\mathbf{P}^{g,c^k} - \alpha\nabla_{\mathbf{P}^{g,c}}\mathcal{L}_\rho(\mathbf{P}^{g,c^k}))$
      /* Using Algorithm 2 */
      S computes $\mathbf{P}^{g,c^k}$ divide by $K$
      /* Using Algorithm 3 */
    end while
    while $c < C$ do
      Update $\boldsymbol{\mu}^{c^{k+1}} = \boldsymbol{\mu}^{c^k} + \rho(\mathbf{P}^{g,c} - \mathbf{P}^{g,c}_o)$
    end while
    Adjust penalty parameter $\rho^c$ when necessary
    $k = k + 1$
  end while
  return $[\theta^c], [\mathbf{P}^{g,c}], c = 0, ..., C$
  CC decrypts $[\mathbf{P}^{g,0}]$ and $[\mathbf{P}^{g,c}]$, divides it by $K$ to get final result.
```

- Furthermore, the system can be easily applied to solve SCOPF multiple times. Assuming that utility companies wish to perform optimization with different settings, it can initiate multiple instances. Consider an ISO want to know their system's operating performance under different numbers of secure constraint. The server can start our systems and testing SCOPF with 20 and 40 secure constraints at the same time.

- Also, multiple estimations can be started when the ISO upload addition secure constraints. In particular, the cryptographic provider does not need to refresh public key too often since the public keys are long-lived, meaning that the ISO can submit more secure constant to the server without changing the previous ones.

4.5.7 Security Analysis

In our model, what we are mainly trying to preserve is the value of $\theta^c, \mathbf{P}^{g,c}, c = 0, ..., C$. Since the computation procedure of ADMM is publicly available, there is no need to hide the computation procedure. We will show the correctness of the algorithms, and then give a proof of security in the honest-but-curious model. For the correctness, we modify the proof of [20].

Definition. The two-party protocol securely computes the function f if there exists two probabilistic polynomial time algorithms for every possible input a, b, f, it is computationally indistinguishable against probabilistic polynomial time adversaries.

$$S_A(a, b, f_A(a, b)) \equiv_c V_A(a, b). \tag{4.55}$$

Here, S_A means all the input of A, $\equiv_c$ means statistically indistinguishable for the adversaries, and V_A means the view of A. Since we do not need to hide the computation procedure, we can reduce the secure definition to if the input and output are statistically indistinguishable for the adversaries.

$$S_A(a, b) \equiv_c V_A(a, b). \tag{4.56}$$

Proposition 1. Algorithm 2 is correct and secure in the honest-but-curious model.

Proof: Since the process of upper and lower projections are identical here, we only try to prove the security of the process of lower boundary projection.

A's view is $V_A = ([x], [s], r, [x < s])$ and the output is $[x_{k+1}], [x - s]^r$. According to [93], it is sufficient to show that there exists a probabilistic polynomial-time algorithm S such that $S(\overline{x}, f(\overline{x}))$ is computationally indistinguishable from V. By semantic security of the Paillier cryptosystem, each encryption are computationally indistinguishable, so this condition is easily verified if S randomly generating r of $\log_c N - 1$ bits. However, the following security is not guaranteed because A will get the $x_k < s$ by comparing $[x_{k+1}]$ with $[s]$. But it does not compromise the privacy of x, since the boundary values s and t are both encrypted.

$$([x], [s], r, (x < s)) \neq_c ([x], [s], r, (x > s)). \tag{4.57}$$

B's view is $V_B = (SK_p, r \cdot (x - s))$ and the output is $r \cdot (x - s) > 0$. So if r_1 and r_2 are taken from the same distribution, independently from any other parameter, $V_B = ((x - s) < 0) = ((x - s) > 0)$. Here, since r is taken randomly in $(-2^l, 2^l)$, the distribution of $(x - s > 0)$ and $(x - s < 0)$ are identical. Therefore, $([x], [s], r, (x < s),)$ and $([x], [s], r, (x > s))$ are statistically indistinguishable. In this case, we conclude that Algorithm 2 is secure in the semi-honest model.

Proposition 2. Algorithm 3 is correct and secure in the honest-but-curious model.

Proof: The view of A consists of its encrypted value $[x]$, and d, its random number r, its output $[(x + r) \div d]$, and all intermediate messages received from B: the encrypted comparison $[c]$. Summarizing, A's view is $V_A = ([x], [z \div d], r)$ and the output is $[x + r]$.

B's view is $V_B = (K_p riv, x + r, z/d)$ and the output is $[x \div d]$. So if r_1 and r_2 are taken from the same distribution, independently from any other parameter. Here, since r is taken randomly in $(-2^l, 2^l)$, the distribution of $r1 \div d$ and $r2 \div d$ are identical. Therefore, follow the similar steps, we conclude with the computational indistinguishability of V_A and V_B.

Proposition 3. Algorithm 4 is correct and secure in the honest-but-curious model.

Proof: Algorithm 4 is semantically secure because the algorithm only calls Algorithm 2 and Algorithm 3, and does not include computations other than the underlying Paillier cryptosystem. Both the called algorithms are semantically secure by our analysis.

Table 4.2: Test cases

Case	$\|\mathcal{N}\|$	$\|\mathcal{G}\|$	$\|\mathcal{B}\|$	Number of Contingency Cases
IEEE 57 buses	57	7	80	50
IEEE 118 buses	118	54	186	100

4.6 Evaluation

In this section, the numerical tests are given to evaluate the performance of the proposed algorithm. Two classical test systems are used the formulate the SCOPF problem: IEEE 57 bus, IEEE 118 bus, whose structures and characteristics are summarized in Table 4.2. Two kinds of contingencies are considered in numerical tests: branch outage and generator failure. The contingencies are artificially generated, and the number of contingencies considered is listed in Table 4.2. We follow the physical limits on the equipment of test systems and assume every active generator is able to reschedule up to 50% of its maximum real power capacity.

The numerical tests are implemented in Java and run on a personal computer with a 2.2GHz processor and 16GB memory. We use the key length of 1024 bits to initialize the Paillier cryptosystem. The basic OPF solver is the same for all test systems. The performance of the convergence and computing time of the proposed algorithm are investigated in the following parts.

4.6.1 Accuracy

One of the most dominant features that affects performance of our system is the number K used in the Paillier cryptosystem. Through synthetic experiments based on IEEE 57 bus case, Table 4.3 illustrates the trade-off between the number of bits of K, with the relative error and time. Suppose $r[k]$ is the result of the value of objective function at the kth iteration, and r^* is the optimal solution. The relative error e is defined as $e = |\frac{r[k]-r^*}{r[0]-r^*}|$. Relative time is defined as $t = |\frac{t[k]-t_{min}}{t_{min}}|$ where t_{min} is the lowest time in the test. It is shown that the larger selection of K will increase the computation time while reducing the relative errors. However, when K is larger than 20 bits, the relative error will significantly increase. This is because the plaintext space of Paillier cryptosystem is overflowed when K is larger

Table 4.3: Trade-offs between the number of bits used for K, relative errors and time

Number of Bites of K	Relative Time	Relative Errors
2	0	0.1409
5	0.0017	0.0389
10	0.2152	0.0247
13	0.2303	0.0073
18	0.2877	0
20	0.3009	1.7819 e+04
25	0.3909	1.7819 e+4

than 20 bits. In this case, for efficient and accuracy consideration, we choose 15 bits as the length of K in the following experiment.

4.6.2 Convergence Rate

We then consider the convergence issue of the proposed algorithm. To better understand the convergence rate between different cases, the relative error is used here to demonstrate the results. The convergence performances are shown in Figure 4.3. It shows that the proposed algorithm converges to the optimal values in all two cases after a few iterations. We can see that with a larger scale of the test system and the number of contingencies, the proposed algorithm has a slower convergence rate, which is due to the fact that a larger system and the number of contingencies considered, lead to a larger optimization problem.

4.6.3 Performance

In this part, we compare the computing time of the proposed algorithm with the centralized approach to solve the SCOPF problem. Note that, both algorithms are performed over the same personal computer. The performance can be further optimized when outsourcing to the cloud like Amazon EC2.

The computing time to obtain the optimal solution is considered in both cases. Communication overhead is also presented. The results of the computing time performance and communication overhead are presented in Table 4.4. From the table, we can see that the computing time of our system is slower than the centralized algorithm. This is because our methods perform entire over the encrypted data. However, compared with the

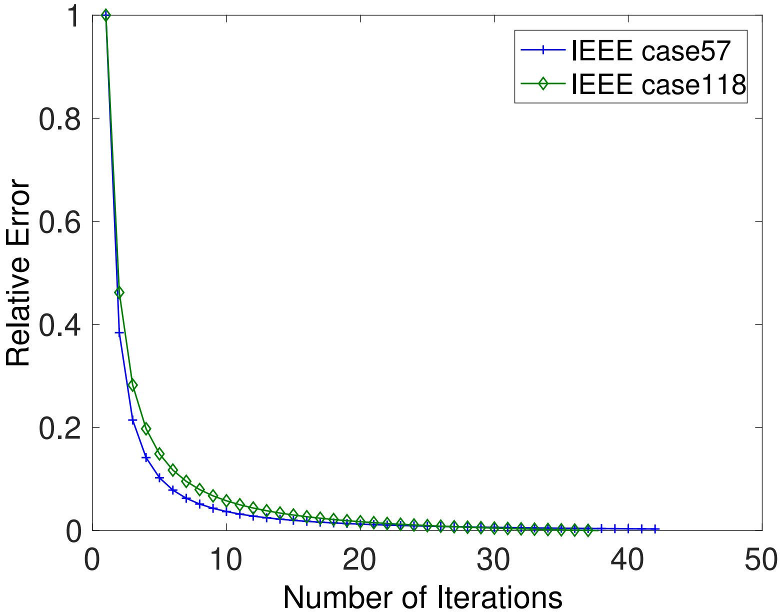


Figure 4.3: Convergence performance

Table 4.4: Computing time of proposed algorithm

Case	**Centralized Algorithm**	**Proposed Algorithm**	
	Time (s)	Comm. (MB)	Time (s)
IEEE 57 buses	5.22	3.59	57.98
IEEE 118 buses	36.03	7.40	208.53

performance of Yao's garbled circuit [88], which is usually 2^{10} time slower than plaintext implementations, our system is significantly more efficient. Besides, the gap between centralized and proposed algorithms becomes smaller when the testing case is larger. This is due to the communication overhead between different processes during the simulations. A larger problem can be achieved on a large-scale test system because the communication overhead is negligible compared with the computing time of the optimization subproblem handled by each computing nodes.

Chapter 5

Privacy-preserving Dynamic Simulations in Power Grid

5.1 Introduction

Dynamical simulations play a critical role in power system research and operations. It is an important approach to accurately predict the dynamic behaviors of power systems under contingencies such as generator tripping, line switching, and short circuit. In the U.S., electricity utilities, regional transmission organizations (RTOs), and independent system operators (ISOs) run dynamic simulations of power system models that are built from large-scale interconnection-wide models with 50,000 or more buses. However, due to the heavy computational burden, dynamic simulations are currently conducted off-line on an hourly or daily basis, limiting their uses in time-critical or real-time power system applications. For example, for a system of comparable size to the Western Electricity Coordinating Council (WECC), commercial power system software takes approximately 60 seconds to simulate 30-second system dynamics following a single contingency using a local workstation with Intel Xeon(R) 3. 2 GHz CPU and 12.0 GB memory. The fact that dynamic simulations take too much time to perform has unfortunately been limiting its online applications in operating modern power systems against a massive list of critical contingencies.

Recently, outsourcing the dynamic simulations to the cloud has emerged as a promising solution to aforementioned problem[76, 59, 77]. Pilot studies conducted by ISO New England

[59][77] showed that by outsourcing the heavy computational burden to the cloud, it is possible to perform power system simulations, not only much faster but also with less cost. According to [77], an N-1-1 contingency analysis with 4,100 scenarios, which would have taken 1,700 hours at a commodity laptop, or 40 hours at the internal computing cluster of 40 cores, now takes a running time of only 1.5 hours with 150 Amazon EC2 nodes for a total monetary cost of about $60. Take another example, it takes around 10 hours to run one study with GE MARS, the Monte Carlo simulation based tool for power system resources adequacy analysis, on a typical commodity laptop. By contrast, with 32 C3.large type of Amazon EC2 instances, the running time comes down to 12 minutes and the cost $1.66.

Despite that outsourcing the dynamical simulations to the public cloud have demonstrated a faster and more cost-effective way to conduct dynamic simulations, potential users like utility companies and ISOs still have reasonable data privacy concerns. It is because the outsourcing also means releasing sensitive and private power system operation data and information to the third-party cloud computing facilities, whose trustability is not always determined. Instead, power companies would opt for conservative approaches such as keeping all the computations local in exchange for the absolute data privacy assurance. Our primary goal of this work is to explore the possibility to outsource the computation without compromising data privacy. The major beneficiaries of our scheme are parties who would like to have the dynamic simulations done in a faster, more economical, and secure way. Those include electrical utilities, regional transmission organizations (RTOs), and independent system operators (ISOs). Cloud-computing service providers may also benefit because the schemes eliminate the user's privacy concern and hence encourages more use of cloud computing service.

5.1.1 Challenges

As mentioned earlier, dynamic simulation is computational extensive and one of the most promising applications in power grid to leverage cloud computing. Similar to the SCOPF problem, the dynamic simulation also contains a set of sensitive parameters which is the primary reason why dynamic simulations are expected to calculate offline.

One obvious way to mitigate this concern is to use secure multi-party computations. However, dynamic simulations are performed at a variety of entities, i.e., utilities, RTOs, and ISOs, which have different requirement and computing power. In this case, it is more practical for a system that has multiple secure level and performance. As such, we are motivated to build a system that has two security level: a provably secure dynamic simulation algorithm with higher computation overhead and an efficient secure dynamic simulation algorithm without mathematical guarantee.

5.1.2 Outlines

The remainder of this chapter is structured as follows. Section 5.2 reviews the related works. Section 5.3 introduces the system model. Background for dynamic simulation is presented in Section 5.4. Privacy-preserving dynamic simulations is covered in Section 5.5. Section 5.6 portrays the evaluations of the proposed algorithm.

5.2 Related Work

The problem of secure outsourcing of scientific computation or performing scientific computation over encrypted data has received considerate attention.

The groundbreaking works by Yao *et al.*[126, 127, 28] demonstrates that any function can be securely computed with garbled circuits. Namely, a user can convert a plain circuit (function) to a garbled circuit and private input to a garbled input; a third-party can then evaluate the garbled circuit with the garbled input to generate the same output as if the original function and input are used. It is guaranteed that the evaluation leaks no more information than the evaluation result. However, the overheads for constructing and evaluating garbled circuits are prohibitively significant for everyday practical computation. More recently, the breakthrough of fully homomorphic encryption[45] allows one to perform unlimited analysis over encrypted data and, thus, appears as the most straightforward solution to outsource of scientific computation. However, like the garbled circuit, even the state-of-the-art fully homomorphic encryption [45] is too inefficient for practical uses.

Another line of research [85, 18, 116, 90, 130] focuses on applying partially homomorphic encryption, or together with garbled circuits, to enable limited processing over encrypted data. Those works, however, focused on securing specific applications. None of them can be applied to solve the problem considered by this chapter, *i.e.* securing the outsourcing of power system dynamical simulations.

Aside from homomorphic encryption and garbled circuit encryption enabled secure outsourcing, it is also reported in the literature that secure outsourcing of scientific computation does not necessarily rely on cryptography [122, 10, 117]. Wang *et al.*[122] proposed secure outsourcing of linear programming. The proposed outsourcing scheme transforms the original linear programming (LP) into a random LP problem. By having the semi-honest cloud solving the random problem and the user transforming the solution back to the original solution, the scheme enables a user to outsource the computation without revealing the original LP problem. Atallah *et al.*[10] also explores the secure outsourcing of scientific computations such as sorting, template matching, string pattern matching by either transforming the original problems into another seemingly different ones or by hiding the original problem in large a collection of similar problems. Nevertheless, unlike cryptography based approaches, those propositions are not provably secure due to lacking a precise definition of security. Our work is based on a provably secure Paillier cryptosystem, and we analyzed the security proposed scheme in Section 5.5.3.

5.3 System Architecture

5.3.1 System Model

Figure 5.1 captures the system model for our scheme. In this chapter, we consider two type of scenarios. For the first scenario, in support of quicker contingency resolution, ISO wishes to get the results of dynamic simulation faster. To that end, the ISO outsources the computations caused by the dynamic simulations to the cloud. However, due to that, the public cloud is an external entity, neither utility companies or ISO fully trusts the cloud; they would like a provably secure way to outsource the computation such that the cloud

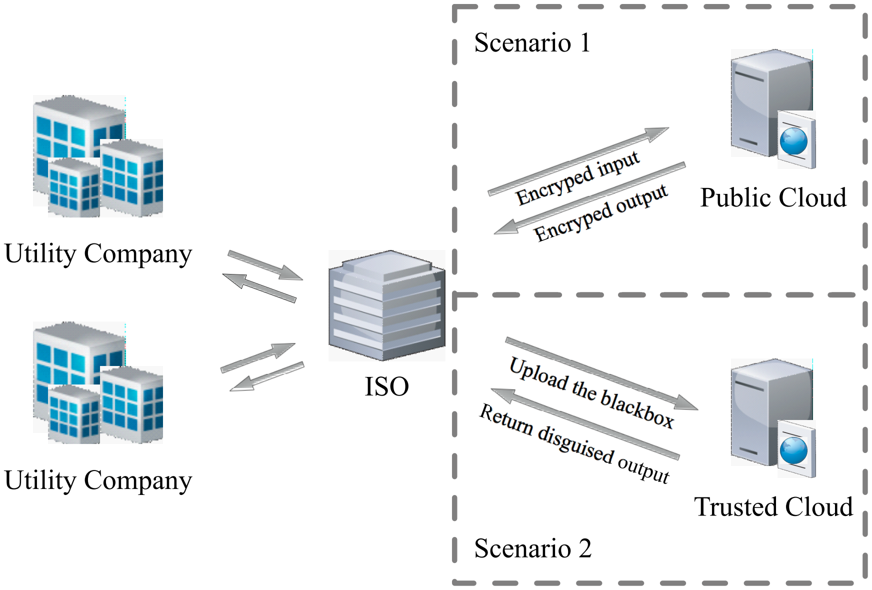


Figure 5.1: Ideal system model for secure computation outsourcing

would assume the computations but without knowing any the private data nor the result of the dynamic simulations. In this case, each utility company encrypt their private date separately and uploaded to the ISO. The ISO then upload encrypted data along with initial conditions to the cloud for computing the dynamic simulation result.

For the second scenario, the ISO wishes to outsource to the cloud power system dynamic simulations with privacy-preservation to the trusted cloud, by uploading a configured black box to the cloud. Here trusted cloud can be private cloud or cloud which are trustworthy. An ideal black box is a piece of executable software that encapsulates all the information it requires, except for the external input, to complete the computation while looking "unintelligible" to the cloud. The cloud performs all computations through the black box. To preserve the privacy of both the input and output values, the black box is constructed in such as way that it takes protected inputs and produce protected outputs, which will be recovered by the user later. This part of the work is done by T. Yue et al. [117]. In this work, we only cover how to secure dynamic simulation in the first scenario, i.e., in the public domain.

As data confidentiality is of the major interest of this work, we enumerate in the following the private data involved in a typical dynamic simulation.

- **Power System Models:** these are the mathematical models, usually non-linear differential algebraic equations, of machines and control systems, such as generators, governors, motor load, *etc.*.

- **Model and contingency data:** the specific parameters and configurations with respect to power system models, including the bus connectivity and information about where and when to apply faults and other disturbances to the system during the simulation period. Model data are usually measured, validated or defined by utility planning engineers.

- **Operating conditions of the system:** the values of system states at the beginning of the simulation. They are updated based on the real-time data from the SCADA (Supervisory Control and Data Acquisition)/EMS (Energy Management System) in the control room.

- **Simulation results:** the output of the contingency simulation in the form of trajectories of state variables over the simulated time.

Data are divided into two categories: input data and output data. Input data are defined as the data submitted to the cloud so that the simulation can be conducted on the cloud end. Output data refer to any information that is generated as the result of the dynamic simulation. Naturally, all kinds of data listed above except for the simulations results belong to the input data. Simulation results are the only kind of output data. Both input data and output data are at risk of unauthorized disclosure if left unprotected.

5.3.2 Design Goals and Threat Model

In this work, we deal with a specific threat that stems from semi-honest cloud service providers, who possess cloud servers that perform the delegated computational tasks and, thus, may knowingly or unknowingly leak private data provided by the data providers, including the utility companies and the ISO in our system model. More specifically, data providers, only partly trust a cloud service provider, in the sense that they believe the cloud service providers will honestly carry out the delegated computation tasks, while they are also aware of that the provider might be curious about and even record any information that pertains to the delegated functions.

The reason for the assumption of the semi-honest adversary is twofold. First, cloud service providers are assumed to honestly perform outsourced tasks because a rational cloud service provider will not do so only if there is a strong incentive and it is certain that not being honest will not be caught. In reality, consider large public cloud providers, for example, Amazon, Microsoft, and Google. For them, the incentive to misconduct pales in comparison with the risk of damaging their reputations. Moreover, providing falsifying computation results can be easily exposed. Second, having the physical control over the cloud servers makes a cloud service provider technically possible to comprise the cloud users' privacy by passively eavesdropping any data that flows in and out the cloud. In practice, the semi-honest model has been increasingly adopted by research work on secure multi-party computation. [121, 85, 90]. Therefore, the assumption of the semi-honest model is reasonable in the setting that the proposed scheme tries to deal with.

The security goal of our work is to limit the adversaries' ability to compromise the users' data confidentiality, which is defined as no private data provided by the utility company are exposed to the cloud. It is worth emphasizing that the confidentiality of the user's data is protected not only when the data is in transit or at rest, but also when they are used in a delegated computation task.

5.4 Preliminaries

In this section, we briefly revisit preliminaries before we present the proposed scheme.

5.4.1 Dynamic Simulations and Runge-Kutta Method

Dynamic simulation refers to using a computer-based approach to study a system's dynamic behavior as a function of time [131]. Dynamic simulation centers around finding the trajectories of state variables by solving the respective initial value problem. In the context of the power system research and operations, dynamic simulations are often performed by numerically solving the differential algebraic equations (DAEs). The resulting trajectories could be examined so as to study the dynamical performance of the system, *e.g.* damping performance of the oscillatory modes w/o additional control scheme and the stability of the under certain disturbances. For instance, to simulate a specific disturbance, the fault-on period is first simulated with a pre-disturbance steady state as the initial condition, and then the last point from the fault-on trajectory is taken as the initial condition to simulate the post-disturbance period.

The classical Runge-Kutta method, a.k.a RK4, may be employed to solve an initial value problem as follows.

$$\dot{y} = f(t, y) \text{ and } y(t_0) = y_0$$

The objective of the initial value problem is to find $y(t)$, which is an unknown function of time. The derivative of y is given by f, a function of time t and y itself. The initial value of y is also given as y_0.

RK4 methods starts with determining a step-size h, so that y's values are approximated every h worth of time along the axis of time. y's value is approximated by the following iterative process.

$$\begin{aligned} y_{n+1} &= y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \\ t_{n+1} &= t_n + h \end{aligned}$$

for $n = 0, 1, 2, 4, ...$ using

$$\begin{aligned} k_1 &= f(t_n, y_n) \\ k_2 &= f(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_1) \\ k_3 &= f(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_2) \\ k_4 &= f(t_n + h, y_n + hk_3) \end{aligned}$$

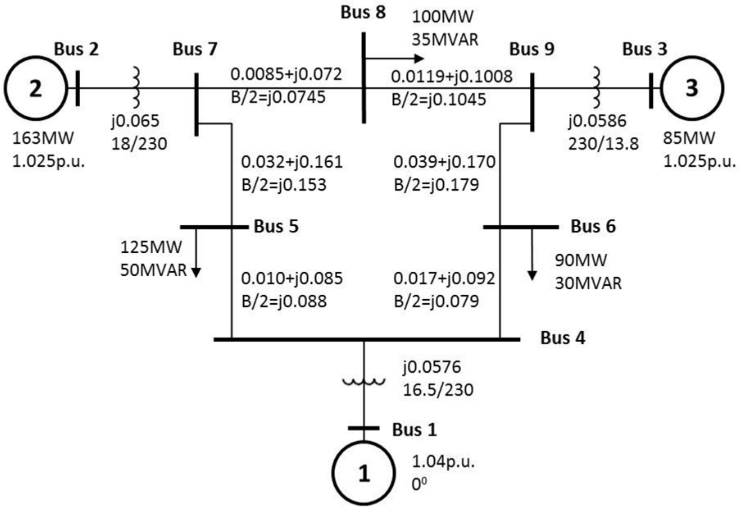


Figure 5.2: IEEE 3-machine, 9-bus system

5.4.2 Paillier Cryptography

Paillier cryptosystem is a provable secure cryptosystem that is based on the decisional composite residuosity assumption [97]. It is widely known as an additive homomorphic cryptography, as it enables one to compute the encryption of the sum of two value, say $m_1 + m_2$, given only the encryption of m_1 and m_2. In this work, we use Paillier cryptosystem which is discribed in Section 4.4.2.

5.4.3 IEEE 3-machine, 9-bus System

The IEEE 3-machine, 9-bus system, as depicted in Figure 5.2, is a simplified model of the Western Electricity Coordinating Council (WECC) system. Without losing generality and for ease of presentation in the following section, we assume that Machine 1 and 2 belong to Area 1 utility and Machine 3 belongs to Area 2 utility. The operation of the whole electrical power grid system is coordinated by a regional ISO.

5.5 Privacy-preserving Dynamical Simulations

In this section, we present how homomorphic encryption is used to secure the outsourcing of dynamic simulations of a particular power system model. We first use the aforementioned exemplary IEEE 3-Machine, 9-Bus system to facilitate the presentation. Next, we generalize the methodology for nonspecific power system models.

5.5.1 Dynamical Simulations for IEEE 3-Machine, 9-Bus System

For the initial discussion and ease of presentation, we consider only the post-disturbance simulation of the contingency: a three-phase fault is added at bus 7 and cleared after five cycles by tripping the Line 5-7 [120].

When the ISO needs to perform the post-disturbance simulation described above, the ISO collect from the utility companies from both areas all inertia constant H_i, all mechanical power P_{mi}, all field voltage E_i, all loads and all transmission line impedances, with which ISO can formulate the admittance matrix.

With all information received from both utility companies, the ISO formulates the following initial value problem to be solved.

$$\dot{x_1} = x_4 \tag{5.1}$$

$$\dot{x_2} = x_5 \tag{5.2}$$

$$\dot{x_3} = x_6 \tag{5.3}$$

$$\dot{x_4} = \pi f / H_1 \cdot (P_{m1} - P_{e1}) \tag{5.4}$$

$$\dot{x_5} = \pi f / H_2 \cdot (P_{m2} - P_{e2}) \tag{5.5}$$

$$\dot{x_6} = \pi f / H_3 \cdot (P_{m3} - P_{e3}) \tag{5.6}$$

The initial condition is given as

$$x_i(t_0) = x_{i0} \text{ for } i = 1, 2, ..., 6$$

where x_i and x_{i+3} are state variables representing the angle and speed of the ith machine, respectively; H_i, P_{mi}, and P_{ei} are the inertia constant, mechanical and electrical power of machine i; f is the synchronous frequency, *i.e.* 50Hz or 60Hz.

Unless a governor is considered, P_{mi} can be regarded as constant, which is the case for this work; The electrical power is calculated as $P_{ei} = \sum_{j=1,2,3} E_i E_j Y_{i,j} cos(\theta_{i,j} - x_i + x_j)$, where $Y_{i,j}$ and $\theta_{i,j}$ are the magnitude and phase of the element in the ith row and jth column of the admittance matrix; E_i is the field voltage of the ith machine. Unless the excitation system is considered, E_i is treated as constant, which is the case in this exemplary model.

Apply the RK4 method to solve the initial value problem. First, we calculate the four increments, as

$$\mathbf{k_1} = \begin{bmatrix} x_4 \\ x_5 \\ x_6 \\ \pi f/H_1 \cdot (P_{m1} - P_{e1}) \\ \pi f/H_2 \cdot (P_{m2} - P_{e2}) \\ \pi f/H_3 \cdot (P_{m3} - P_{e3}) \end{bmatrix} \tag{5.7}$$

$$\mathbf{k_2} = \begin{bmatrix} x_4 + \frac{h}{2}x_4 \\ x_5 + \frac{h}{2}x_5 \\ x_6 + \frac{h}{2}x_6 \\ \pi f/H_1 \cdot (P_{m1} - P_{e1}) \\ \pi f/H_2 \cdot (P_{m2} - P_{e2}) \\ \pi f/H_3 \cdot (P_{m3} - P_{e3}) \end{bmatrix} \tag{5.8}$$

$$\mathbf{k_3} = \begin{bmatrix} x_4 + \frac{h}{2}(x_4 + \frac{h}{2}x_4) \\ x_5 + \frac{h}{2}(x_5 + \frac{h}{2}x_5) \\ x_6 + \frac{h}{2}(x_6 + \frac{h}{2}x_6) \\ \pi f/H_1 \cdot (P_{m1} - P_{e1}) \\ \pi f/H_2 \cdot (P_{m2} - P_{e2}) \\ \pi f/H_3 \cdot (P_{m3} - P_{e3}) \end{bmatrix} \tag{5.9}$$

$$\mathbf{k_4} = \begin{bmatrix} x_4 + h(\frac{h}{2}(x_4 + \frac{h}{2}x_4)) \\ x_5 + h(\frac{h}{2}(x_5 + \frac{h}{2}x_5)) \\ x_6 + h(\frac{h}{2}(x_6 + \frac{h}{2}x_6)) \\ \pi f/H_1 \cdot (P_{m1} - P_{e1}) \\ \pi f/H_2 \cdot (P_{m2} - P_{e2}) \\ \pi f/H_3 \cdot (P_{m3} - P_{e3}) \end{bmatrix} \tag{5.10}$$

Since,

$$\mathbf{x_{n+1}} = \mathbf{x_n} + \frac{h}{6}(\mathbf{k_1} + 2\mathbf{k_2} + 2\mathbf{k_3} + \mathbf{k_4}) \tag{5.11}$$

Therefore,

$$\mathbf{x_{n+1}} = \begin{bmatrix} x_1 + x_4(1 + \frac{1}{3}h + \frac{1}{6}h^2 + \frac{h^3}{24}) \\ x_2 + x_5(1 + \frac{1}{3}h + \frac{1}{6}h^2 + \frac{h^3}{24}) \\ x_3 + x_6(1 + \frac{1}{3}h + \frac{1}{6}h^2 + \frac{h^3}{24}) \\ x_4 + h\pi f/H_1 \cdot (P_{m1} - P_{e1})) \\ x_5 + h\pi f/H_2 \cdot (P_{m2} - P_{e2})) \\ x_6 + h\pi f/H_3 \cdot (P_{m3} - P_{e3})) \end{bmatrix} \tag{5.12}$$

Note that h is the step size in RK-4 and is thus constant. Therefore, we denote a constant $C1$

$$C_1 = 1 + \frac{1}{3}h + \frac{1}{6}h^2 + \frac{h^3}{24}$$

.

Although $P_{ei} = \sum_{j=1,2,3} E_i E_j Y_{i,j} cos(\theta_{i,j} - x_i + x_j)$ is a nonlinear function of x_i, its value can be calculated, as the present value of x_i are known in the beginning of a iteration, and the value remains constant throughout the iteration. Moreover, f, H_1, H_2, H_3, and P_{m1}, P_{m2}, P_{m3} are all constant. As such, we may denote additional constants C_{2i} and

$$C_{2i} = h\pi f/H_i \cdot (P_{mi} - P_{ei}) \quad \text{for} \;\; i = 1, 2, 3$$

As a result of the new denotations, we may rewrite $\mathbf{x_{n+1}}$ as

$$\mathbf{x_{n+1}} = \begin{bmatrix} x_1 + x_4 C_1 \\ x_2 + x_5 C_1 \\ x_3 + x_6 C_1 \\ x_4 + C_{21} \\ x_5 + C_{22} \\ x_6 + C_{23} \end{bmatrix} \tag{5.13}$$

As seen, it turns out that the next values of the state variables, $\mathbf{x_{n+1}}$, are a linear function of the present values, when P_{ei}'s are treated as constants. It follows that, because of Paillier encryption's additivity, given the current values of x_i's Paillier encryption, for $i = 1, 2, .., 6$, and constant values of C_1 and C_2, one can calculate the Paillier encrypted next values of x_i, for $i = 1, 2, .., 6$.

Let $Enc(\cdot)$ denote a Paillier encryption operator and $||\cdot||$ the respective ciphertext. Recall that Paillier encryption's the additive homomorphism and its homomorphic multiplication by a plaintext that are covered in Section. 5.4. It follows that, $Enc(\mathbf{x_{n+1}})$ is equivalent to,

$$\begin{bmatrix} Enc(x_{1,n+1}) \\ Enc(x_{2,n+1}) \\ Enc(x_{3,n+1}) \\ Enc(x_{4,n+1}) \\ Enc(x_{5,n+1}) \\ Enc(x_{6,n+1}) \end{bmatrix} = \begin{bmatrix} Enc(x_1) \cdot Enc(x_4)^{C_1} \\ Enc(x_2) \cdot Enc(x_5)^{C_1} \\ Enc(x_3) \cdot Enc(x_6)^{C_1} \\ Enc(x_4) \cdot Enc(C_{21}) \\ Enc(x_5) \cdot Enc(C_{22}) \\ Enc(x_6) \cdot Enc(C_{23}) \end{bmatrix} \tag{5.14}$$

As such, the ISO can securely outsource the initial value problem described in Equation 5.2 through 5.6 with Algorithm 5 and Algorithm 6.

5.5.2 Privacy-preserving RK4 Protocol

In this subsection, we generalize the above processes, which works for a specific power system model, in order to securely outsource dynamical simulations to the cloud in general. We present the generalized method as a protocol that consists of four phases, as shown in Figure 5.3.

- **Preparation Phase** The ISO server bootstraps the Paillier cryptosystem by taking a set of security parameter, generating the private-key and public-key pair. The ISO server then distributes the public key to all cloud server instances it has started and all utility companies that are involved in the dynamical simulations that need to perform.

- **Phase 1** Each utility company u_i involved in the dynamical simulation conducts the power flow studies locally to obtain its share of the information necessary for converting the dynamic simulation to an initial value problem. This information may

Algorithm 5 Secure outsourcing IVPs for IEEE 3-Machine 9-Bus system to public cloud

```
Input: an IVP defined with Eq. 5.2-5.6
  $h$
  $t_{end}$
Output: Trajectories of state variables $C_1 \leftarrow (1 + \frac{1}{3}h + \frac{1}{6}h^2 + \frac{h^3}{24})\$;$
  for $i$ in $\{1, 2, ..., 6\}$ do
    $x_i \leftarrow x_{i0}$
  end for
  while $t_n < t_{end}$ do
    for $i$ in $\{1, 2, 3\}$ do
      $C_{2i} \leftarrow h\pi f/H_i \cdot (P_{mi} - \sum_{j=1,2,3} E_i E_j Y_{i,j} cos(\theta_{i,j} - x_i + x_j))$ /* Update $C_{2i}$ */
    end for
    for $i$ in $\{1, 2, ..., 6\}$ do
      $||x_i|| \leftarrow Enc(x_i)$ /* Re-encrypt current values of $x_i$ */
    end for
    $||\mathbf{x_n}|| \leftarrow ||x_1||, ||x_2||, ||x_3||, ||x_4||, ||x_5||, ||x_6||$
    $||\mathbf{x_{n+1}}|| \leftarrow$ OutsourceToCloud ($||\mathbf{x}||$,$C_1$,$||C_{2i}||$) /* Outsource the iteration to the cloud*/

    $||x_{1,n+1}||, ||x_{2,n+1}||, ...||x_{6,n+1}|| \leftarrow ||\mathbf{x_{n+1}}||$
    for $i$ in $\{1, 2, ..., 6\}$ do
      /* Decrypt and update current values */
      $x_i(t_n) \leftarrow x_i$
      $x_i \leftarrow Dec(||x_{i,n+1}||)$
    end for
    $t_n \leftarrow t_n + h$
  end while
  Return $x_i(t_0 + kh)$ for $i \in [1, 6]$ and $k$ such that $t_0 + kh \leq t_{end}$
```

Algorithm 6 OutsourceToCloud (executed on the cloud servers)

```
Input: $||\mathbf{x_n}||$,$C_1$,$||C_{2i}||$
Output: $||\mathbf{x_{n+1}}||$
  $||x_1||, ||x_2||, ..., ||x_6|| \leftarrow ||\mathbf{x_n}||$
  for $i$ in $\{1, 2, 3\}$ do
    $||x_{i,n+1}|| \leftarrow ||x_i|| \cdot ||x_{i+3}||^{C_1}$
  end for
  for $i$ in $\{1, 2, 3\}$ do
    $||x_{3+i,n+1}|| \leftarrow ||x_i|| \cdot ||C_{2i}||$
  end for
  $||\mathbf{x_{n+1}}|| \leftarrow ||x_{1,n+1}||, ||x_{2,n+1}||, ...||x_{6,n+1}||$
  Return $||\mathbf{x_{n+1}}||$
```

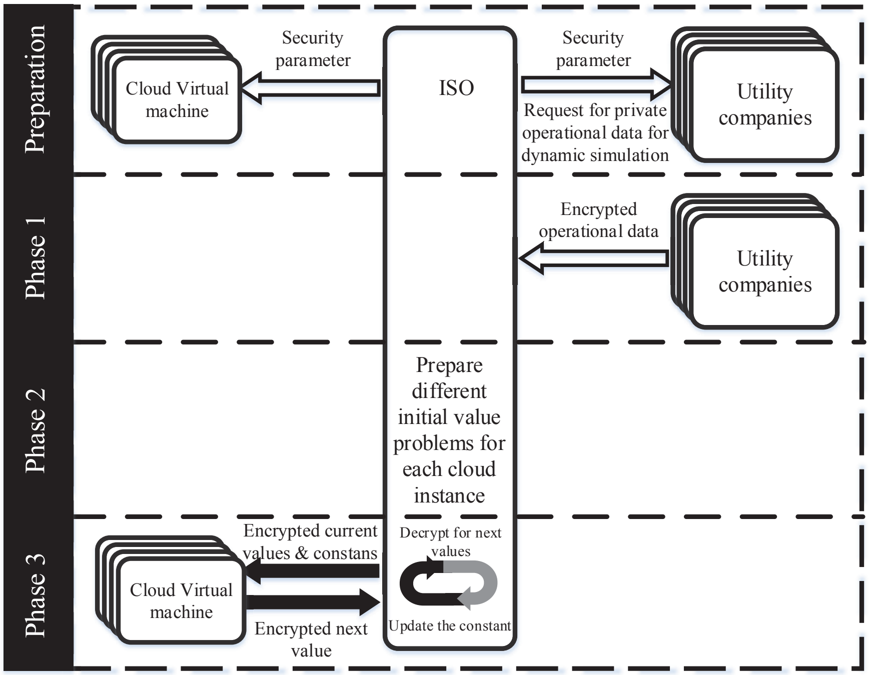


Figure 5.3: Description of proposed protocol.

include inertia constants, mechanical power of all machines, field voltages, loads and transmission line impedances, *etc.*. Each utility company then sends the information to the ISO server through a secure channel between the utility company and the ISO established with the private and public key pair.

- **Phase 2** Upon receiving the information from all concerned utility companies, the ISO server performs necessary processing to generate the dynamical simulations and the corresponding initial value problems. Each IVP is then to be outsourced to a cloud server employed by the ISO. As our goal is to securely outsource part of the computation to the cloud server, for each IVP, the ISO server needs to pick out the non-linear operation that cannot be calculated homomorphically by Paillier encryption and performs them locally. For instance, $P_{ei} = \sum_{j=1,2,3} E_i E_j Y_{i,j} cos(\theta_{i,j} - x_i + x_j)$ for $i = 1, 2, 3$ in the above example. Following that, the ISO server also combines the constant terms, if any, into a single term like C_1 in the above example. In addition, the ISO server calculates the coefficient for the current values in an iteration of RK4 method, like $C_{2i} = h\pi f / H_i \cdot (P_{mi} - P_{ei})$ for $i = 1, 2, 3$. Note that for a particular model, the determinations of which non-linear operation must be carried out, constant terms need to combine, and coefficient to calculate out before each iteration can be done once and for all. Constant coefficients or terms like C_1 in the above example can also be used across all iterations.

- **Phase 3** This is the only phase that requires both the ISO server and the cloud server to be online at the same time. Once an iteration of the RK4 integration is expressed such the next values of the unknown function are linear functions of present values, the initial value problem can be securely outsourced to the public cloud. In support of outsourcing, the ISO server encrypts the current values of the state variables and sends the ciphertext to the cloud server, along with the constants in plaintext like C_1 and C_{2i}. With the information received from the ISO server, the cloud server can homomorphically calculate the ciphertexts of the next values of the state variables. The cloud server returns the ciphertexts to the ISO server, which then uses the public key to decrypt for the next values in plaintext. Next, the ISO server saves the actual subsequent values of the state variable as part of the final solution of the IVP. The ISO server then updates the current state variables' value with the next ones and also updates the constant terms and coefficients as necessary. With the new current values and updated values of the constant terms, the ISO server initiates another iteration of the secure outsourcing. In the same manner, iterations continue till the end of the simulated period is reached.

5.5.3 Discussion

In this subsection, we examine the correctness and security of our algorithms in the semi-honest model [47].

The correctness property is straightforward which underlies the correctness of Paillier cryptosystem. However, since Paillier cryptosystem only works for integers, we have to introduce a constant k (e.g.$k = 2^{52}$ for IEEE 754 doubles) and multiply it with each floating point value before each operation and thus support finite precision. The selection of k will inevitably introduce error into the dynamic simulation. Nonetheless, the experimental result shows that the relative error is on the order of 10^{-6} which is analyzed in the next section.

Then let's consider the security of our algorithm. Recall that in the semi-honest model, the parties honestly follow the protocol but may motivate to misbehave and potentially learn the data provided by the service initiator. To guarantee security in the semi-honest model, we must show that parties can learn nothing beyond their outputs from the information they get throughout the entire process. In our system, the cloud can only view ciphertexts from the ISO and from iterative outputs and the public key K_{pub} which it may use to generate any number of ciphertexts. First of all, in cryptography, the privacy of the data is preserved when it is processed over an algorithm that is semantic secure.

Definition 1: A cryptosystem is **semantic secure** if given a ciphertext encrypted from a known plaintext set (m_1,m_2), the adversary cannot determine which of the two plaintexts was encrypted, with probability significantly greater than 1/2.

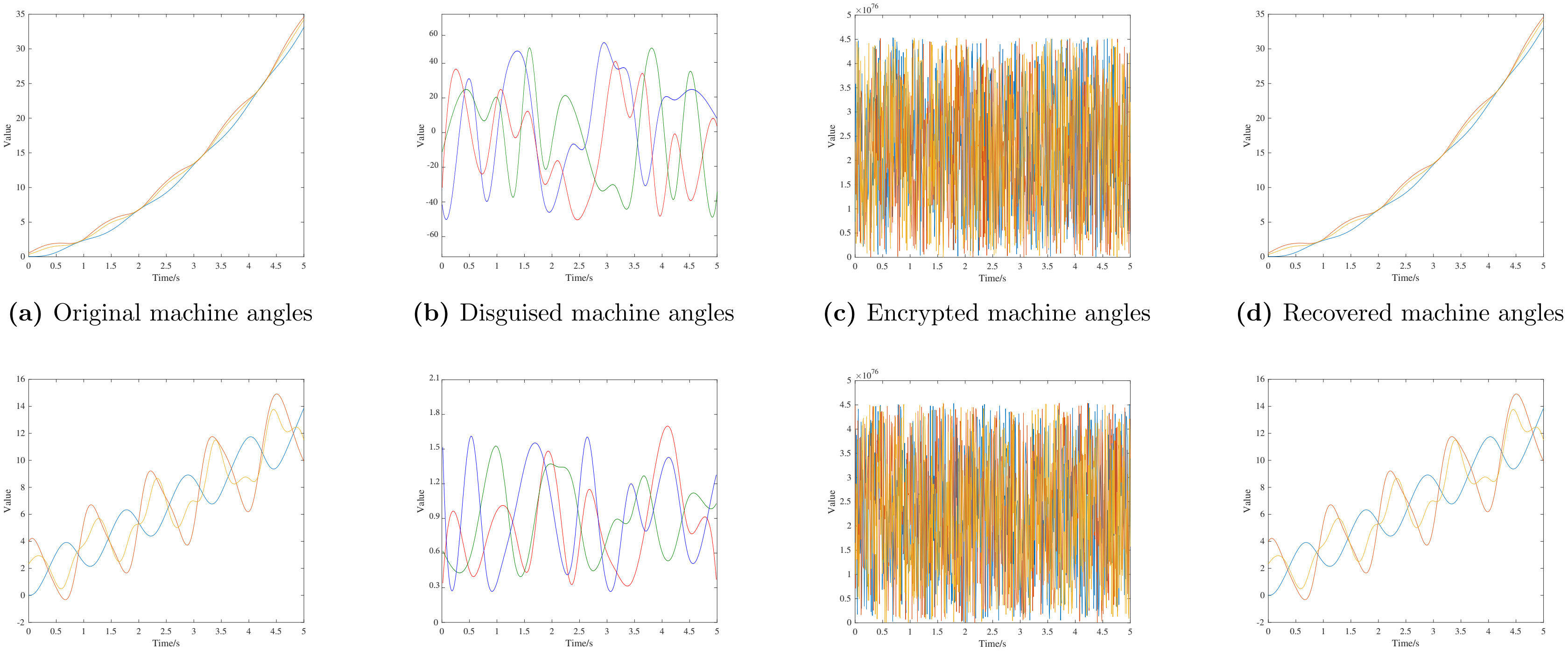
Paillier cryptosystem which used in our system is proved to have semantic security in [97]. Although the cloud undertakes most of the computations, all the operations are carried over the ciphertext encrypted under the Paillier public key. In this case, our algorithm is semantically secure because the cloud does not learn intermediary results in the computation, because of the security of the Paillier cryptosystem, and because it gets a refreshed ciphertext from the server which the cloud cannot couple to a previously seen ciphertext.

5.6 Evaluation

5.6.1 Implementation

We implemented the proposed secure dynamic simulation scheme using the IEEE 3-Machine 9-Bus System. The numerical tests are implemented via C++ on a regular laptop with an Intel 2.20GHz and 16Gb memory. The program uses GNU Multiple Precision Arithmetic Library in the implementation of Paillier cryptosystem. Step size is set to fix 1/120 second. For Paillier cryptosystem, we use the key length of 128 bits. The cloud and server run on different Virtual Machines, each using a single processor. The performance of the computing time and communication overhead of the proposed algorithm are investigated in the following parts.

Figure 5.4 shows the original, disguised, encrypted, and recovered trajectories of the generators' angles and speeds. It is shown that the original and the recovered trajectories are identical. Taking the difference between the two, which is a zero vector, verifies our observation. Thereby, the outsourcing scheme can disguise the original result and also recover the disguised results without any error.



(a) Original machine angles

(b) Disguised machine angles

(c) Encrypted machine angles

(d) Recovered machine angles

(e) Original machine speeds

(f) Disguised machine speeds

(g) Encrypted machine speeds

(h) Recovered machine speeds

Figure 5.4: Original, disguised, encrypted, and recovered curves of machine angles and machine speeds

5.6.2 Performance

We now analyze the computation and communication overhead of our algorithm. More specifically, we measure the overall time for privacy computation and the amount of data moving across parties in the entire process.

Phase 1

The performance of Phase 1 depends on the problem size. For instance, a typical 3 machine 9 bus model will incur a one-time 674 bytes communication overhead between utility companies and the ISO.

Phase 2

Next, the implementation of Phase 2 using Paillier's scheme consists of two part: the computation overhead to perform Paillier encryption and the communication overhead. Based on our experiments, the Paillier encryption will incur a 0.01 s computation overhead for IEEE 3 machine 9 bus system. This overhead will possibly increase when we solve larger problems. However, this overhead is one-time and can be further reduced by parallelization or using specialized hardware, such as FPGA, optimized for cryptographic operations.

Phase 3

To better understanding the trade-off for adding privacy-preserving, we first consider the execution time of non-privacy-preserving dynamic simulation algorithm. As shown in Table 5.1, computation cost, and communication overhead are listed. From the table, we can see that our algorithm is slower than the plaintext implement due to the added security guarantee. However, given the same computation resource, our algorithm is only less than 10^2 times slower with a reasonable amount of communication overhead.

Since we are the first to study privacy preserving power system simulations, no other algorithm is available for similar problems. A generic solution to perform operations over encrypted data is to use Yao's garbled circuit. However, given a 10^4 to 10^6 times [72]

Table 5.1: Computing time and communication overhead of proposed algorithm

Time Span	**Non-privacy-preserving Approach**	**Proposed Algorithm**	
	Time (s)	Comm. (MB)	Time (s)
1s	0.0049	0.45	0.47
5s	0.011	2.11	2.47
15s	0.022	6.88	6.94
30s	0.034	13.545	13.41

slowdown and the real-time requirement for dynamic simulation, for practical purposes, our algorithm is significantly more suitable than garbled circuit based methods.

5.6.3 Accuracy

Since our proposed system implements dynamic simulation in a numerical way with fixed-point numbers, it will inevitably introduce errors. Denote by $\mathbf{x}^*$ the solution in the plaintext using Matlab on a 64bit commodity server, and $\mathbf{x}$ to be the solution using our system. We define the relative error as:

$$e = \|\frac{\mathbf{x} - \mathbf{x}^*}{\mathbf{x}^*}\|_2^2$$

Through a broad of experiments, k in our system is expected to be 10^6, and the cloud seeks an error on the order of 10^{-6}. Although the cloud can decide to use a larger k to lower the error, however, it will raise the risk that the crypto space being overflowed.

Chapter 6

Conclusions and Future Works

In this dissertation, we investigate vulnerability assessment and privacy-preserving computations in the smart grid.

6.1 Conclusions

In Chapter 2, we present the vulnerability assessment over the synchrophasor network. Unlike previous works, our work primarily focuses on testing the effeteness of cyber attacks in real-world scenarios. Specifically, three practical cyber attack schemes (DoS attack, sniffing attack, false data injection attack) over synchrophasor networks is demonstrated on CURENT Hardware. We also discussed how each cyber attack could impact the synchrophasor network based Wide-area monitoring system (WAMS). In addition, symmetric-key algorithm and keyed-hash message authentication code (HMAC) for message encryption, integrity verification, and authentication are tested over C37.118. We also carry out practical recommendations to secure synchrophasor networks for different attacks including cryptographic techniques, key management, intrusion detection systems, etc.

In Chapter 3, we propose a deep learning based framework to detect measurement anomalies due to False Data Injection (FDI) attacks. We described our detection methodology that leverages both convolutional neural network and recurrent neural networks. Our model learns normal behavior from normal data and is not restricted to FDI attacks, and thus can detect other unseen attacks. Additionally, our two-level detector is robust using hybrid features and can detect the attack when state vector estimator fails. We provided critical insights into various factors that impact the performance of the proposed algorithm. We presented a detailed case study of the proposed algorithm on the IEEE 39-bus system.

In Chapter 4, the original SCOPF problem is decoupled and divided into small subproblems. The subproblems are approximately the same size and optimized in a parallel fashion on distributed nodes. We then presented a practical approach that can solve the large-scale SCOPF problem directly over the encrypted SCOPF problem to guarantee the security and privacy of the system. The privacy-preserving ADMM and gradient projection algorithms were also proposed to support the scheme. The scheme is computationally secure and parallelizable based on additive homomorphic cryptosystem. The numerical tests on IEEE buses were carried out, which showed that our proposed scheme is less than 2^4 times slower than the non-privacy-preserving method. Moreover, security analysis proves that our algorithm can preserve both system confidentiality and data privacy against semi-honest attackers. As a result, the SCOPF problem can be solved by entities with abundant computational resources to achieve better efficiency and economy using our scheme.

In Chapter 5, a novel scheme was proposed to securely outsource nonlinear power system dynamical simulations to the semi-honest public cloud. The goal of the proposition was to protect the data privacy of the data providers, *i.e.* ISO and utility companies, when the ISO offloads part of the computation burden to a not fully trusted cloud, so that ISO or utility companies may take advantage of the lower cost, agility, scalability, and on-demand provisioning offered by the cloud computing, without the risk of data breach. The security of the proposed scheme was analyzed. The proposed scheme was implemented and tested with the IEEE 3-Machine 9-Bus System. The performance of the implementation was evaluated and compared with that of Yao's garbled circuit based approaches.

6.2 Future Research Directions

Based on the work to date, continuing research in the following direction is needed:

- Addition cyber attacks against other communication networks in smart grid should be tested. For instance, in Substation Automation System (SAS), traditional electronic devices at substations have been upgraded to intelligent electronic devices (IEDs). With IEDs, power system operators are able to monitor and control a power system from a remote control center.

- The redundancy of and how easy to get measurement Jacobian matrix **H** for false data inject attacks should be examined. We also want to see if adversaries can generate **H** from other power system data.

- It is helpful to extend the idea of using homomorphic cryptosystem to build more applications. For example, Prony analysis is an emerging methodology that extends Fourier analysis by directly estimating the frequency, damping, and strength for a given signal which is widely adopted in power system monitoring and controls.

Bibliography

[1] (2018). Hack Back: a DIY guide. http://pastebin.com/raw/0SNSvyjJ. Accessed: 2018-09-20. 6

[2] Richard J. Campbell (2018). The smart grid: Status and outlook. https://fas.org/sgp/crs/misc/R45156.pdf. Accessed: 2018-08-25. 1

[3] Verizon (2018). 2018 data breach investigations report. https://www.verizonenterprise.com/resources/reports/rp_DBIR_2018_Report_execsummary_en_xg.pdf. Accessed: 2018-08-25. 6

[4] Agrawal, R. and Srikant, R. (2000). Privacy-preserving data mining. In *ACM Sigmod Record*, volume 29, pages 439–450. 68

[5] Almas, M. S., Vanfretti, L., Singh, R. S., and Jonsdottir, G. M. (2018). Vulnerability of synchrophasor-based wampac applications' to time synchronization spoofing. *IEEE Transactions on Smart Grid*, 9(5):4601–4612. 15

[6] Alsac, O. and Stott, B. (1974). Optimal load flow with steady-state security. *IEEE Transactions on Power Apparatus and Systems*, PAS-93(3):745–751. 63, 64

[7] Anwar, A., Mahmood, A. N., and Tari, Z. (2015). Identification of vulnerable node clusters against false data injection attack in an ami based smart grid. *Information Systems*, 53:201–212. 25

[8] Ardakani, A. J. and Bouffard, F. (2013). Identification of umbrella constraints in dc-based security-constrained optimal power flow. *IEEE Transactions on Power Systems*, 28(4):3924–3934. 67

[9] Ashok, A., Govindarasu, M., and Ajjarapu, V. (2016). Online detection of stealthy false data injection attacks in power system state estimation. *IEEE Transactions on Smart Grid*. 46, 56

[10] Atallah, M. J., Pantazopoulos, K. N., Rice, J. R., and Spafford, E. H. (2001). Secure outsourcing of scientific computations. *Advances in Computers*, 54:215–272. 95

[11] Baharlouei, Z. and Hashemi, M. (2014). Efficiency-fairness trade-off in privacy-preserving autonomous demand side management. *IEEE Transactions on Smart Grid*, 5(2):799–808. 7, 67

[12] Barker, E., Smid, M., Branstad, D., and Chokhani, S. (2013). A framework for designing cryptographic key management systems. *NIST Special Publication*, 800:130. 38

[13] Beasley, C., Zhong, X., Deng, J., Brooks, R., and Venayagamoorthy, G. K. (2014). A survey of electric power synchrophasor network cyber security. In *Innovative Smart Grid Technologies Conference Europe (ISGT-Europe), 2014 IEEE PES*, pages 1–5. IEEE. 5, 8, 14, 15, 37

[14] Bellare, M., Hoang, V. T., Keelveedhi, S., and Rogaway, P. (2013). Efficient garbling from a fixed-key blockcipher. In *Security and Privacy (SP), 2013 IEEE Symposium on*, pages 478–492. 82

[15] Bengio, Y., Simard, P., and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166. 52

[16] Bertsekas, D. P. (1999). *Nonlinear programming*. Athena scientific Belmont. 65

[17] Bi, S. and Zhang, Y. J. (2014). Graphical methods for defense against false-data injection attacks on power system state estimation. *IEEE Transactions on Smart Grid*, 5(3):1216–1227. 46, 56

[18] Bianchi, T., Piva, A., and Barni, M. (2009). On the implementation of the discrete fourier transform in the encrypted domain. *Information Forensics and Security, IEEE Transactions on*, 4(1):86–97. 95

[19] Bobba, R. B., Rogers, K. M., Wang, Q., Khurana, H., Nahrstedt, K., and Overbye, T. J. (2010). Detecting false data injection attacks on dc state estimation. In *Preprints of the First Workshop on Secure Control Systems, CPSWEEK*, volume 2010. 46

[20] Bost, R., Popa, R. A., Tu, S., and Goldwasser, S. (2014). Machine learning classification over encrypted data. *IACR Cryptology ePrint Archive*, 2014:331. 82, 87

[21] Boyd, S., Parikh, N., Chu, E., Peleato, B., and Eckstein, J. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122. 65

[22] Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press, New York, NY, USA. 74

[23] Byres, E. and Lowe, J. (2004). The myths and facts behind cyber security risks for industrial control systems. In *Proceedings of the VDE Kongress*, volume 116, pages 213–218. 21

[24] Capitanescu, F., Glavic, M., Ernst, D., and Wehenkel, L. (2007). Contingency filtering techniques for preventive security-constrained optimal power flow. *IEEE Transactions on Power Systems*, 22(4):1690–1697. 67

[25] Capitanescu, F., Ramos, J. M., Panciatici, P., Kirschen, D., Marcolini, A. M., Platbrood, L., and Wehenkel, L. (2011). State-of-the-art, challenges, and future trends in security constrained optimal power flow. *Electric Power Systems Research*, 81(8):1731 – 1741. 64

[26] Cárdenas, A. A., Amin, S., Schwartz, G., Dong, R., and Sastry, S. (2012). A game theory model for electricity theft detection and privacy-aware control in ami systems. In *Communication, Control, and Computing (Allerton), 2012 50th Annual Allerton Conference on*, pages 1830–1837. IEEE. 47

[27] Carpentier, J. (1962). Contribution a l'etude du dispatching economique. *Bulletin de la Societe Francaise des Electriciens*, 3(1):431–447. 63

[28] Chaum, D., Crépeau, C., and Damgard, I. (1988). Multiparty unconditionally secure protocols. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, STOC '88, pages 11–19, New York, NY, USA. ACM. 94

[29] Chen, J. and Abur, A. (2006). Placement of pmus to enable bad data detection in state estimation. *IEEE Transactions on Power Systems*, 21(4):1608–1615. 25

[30] Chu, Z., Zhang, J., Kosut, O., and Sankar, L. (2017). Vulnerability assessment of large-scale power systems to false data injection attacks. *arXiv preprint arXiv:1705.04218*. 44

[31] crowdstrike (2018). Cyber intrusion services casebook 2017. ww.crowdstrike.com/resources/reports/cyber-intrusion-services-casebook. Accessed: 2018-08-15. 6

[32] Dagle, J. E. (2010). The north american synchrophasor initiative (naspi). In *Power and Energy Society General Meeting, 2010 IEEE*, pages 1–3. IEEE. 17

[33] Damgård, I. and Jurik, M. (2001). A generalisation, a simpli. cation and some applications of paillier's probabilistic public-key system. In *International Workshop on Public Key Cryptography*, pages 119–136. Springer. 74

[34] Deng, R., Xiao, G., Lu, R., Liang, H., and Vasilakos, A. V. (2017). False data injection on state estimation in power systems—attacks, impacts, and defense: A survey. *IEEE Transactions on Industrial Informatics*, 13(2):411–423. 25

[35] Deng, W. and Yin, W. (2016). On the global and linear convergence of the generalized alternating direction method of multipliers. *Journal of Scientific Computing*, 66(3):889–916. 85

[36] Du, P. and Makarov, Y. (2014). Using disturbance data to monitor primary frequency response for power system interconnections. *IEEE Transactions on Power Systems*, 29(3):1431–1432. 17

[37] Engel, D. and Eibl, G. (2017). Wavelet-based multiresolution smart meter privacy. *IEEE Transactions on Smart Grid*, 8(4):1710–1721. 7, 67

[38] Esmalifalak, M., Liu, L., Nguyen, N., Zheng, R., and Han, Z. (2014). Detecting stealthy false data injection using machine learning in smart grid. *IEEE Systems Journal*. 44, 56

[39] Evans, D., Huang, Y., Katz, J., and Malka, L. (2011). Efficient privacy-preserving biometric identification. In *Proceedings of the 17th conference Network and Distributed System Security Symposium, NDSS*. 68

[40] Fadlullah, Z. M., Fouda, M. M., Kato, N., Shen, X., and Nozaki, Y. (2011). An early warning system against malicious activities for smart grid communications. *IEEE Network*, 25(5). 39

[41] Falliere, N., Murchu, L. O., and Chien, E. (2011). W32. stuxnet dossier. *White paper, Symantec Corp., Security Response*, 5(6):29. 13, 36

[42] Feng, C., Li, T., and Chana, D. (2017). Multi-level anomaly detection in industrial control systems via package signatures and lstm networks. In *Dependable Systems and Networks (DSN), 2017 47th Annual IEEE/IFIP International Conference on*, pages 261–272. IEEE. 58

[43] Fouda, M. M., Fadlullah, Z. M., Kato, N., Lu, R., and Shen, X. S. (2011). A lightweight message authentication scheme for smart grid communications. *IEEE Transactions on Smart Grid*, 2(4):675–685. 43

[44] Gellings, C. (2011). Estimating the costs and benefits of the smart grid: a preliminary estimate of the investment requirements and the resultant benefits of a fully functioning smart grid. *Electric Power Research Institute (EPRI), Technical Report (1022519)*. 1

[45] Gentry, C. (2009). *A fully homomorphic encryption scheme*. PhD thesis, Stanford University. 94

[46] Giaconi, G., Gündüz, D., and Poor, H. V. (2018). Smart meter privacy with renewable energy and an energy storage device. *IEEE Transactions on Information Forensics and Security*, 13(1):129–142. 7, 67

[47] Goldreich, O. (2004). *Foundations of Cryptography: Volume 2, Basic Applications*. Cambridge University Press, New York, NY, USA. 71, 108

[48] Gomez-Exposito, A. and Abur, A. (2004). *Power system state estimation: theory and implementation*. CRC press. 39, 49

[49] Gong, Y., Cai, Y., Guo, Y., and Fang, Y. (2016). A privacy-preserving scheme for incentive-based demand response in the smart grid. *IEEE Transactions on Smart Grid*, 7(3):1304–1313. 7, 67

[50] Greer, C., Wollman, D. A., Prochaska, D. E., Boynton, P. A., Mazer, J. A., Nguyen, C. T., FitzPatrick, G. J., Nelson, T. L., Koepke, G. H., Hefner Jr, A. R., et al. (2014). Nist framework and roadmap for smart grid interoperability standards, release 3.0. Technical report. 2, 4

[51] Hahn, A., Ashok, A., Sridhar, S., and Govindarasu, M. (2013). Cyber-physical security testbeds: Architecture, application, and evaluation for smart grid. *IEEE Transactions on Smart Grid*, 4(2):847–855. 15

[52] Hao, Y., Wang, M., and Chow, J. H. (2016). Likelihood analysis of cyber data attacks to power systems with markov decision processes. *IEEE Transactions on Smart Grid*. 58

[53] He, Y., Mendis, G. J., and Wei, J. (2017). Real-time detection of false data injection attacks in smart grid: A deep learning-based intelligent mechanism. *IEEE Transactions on Smart Grid*, 8(5):2505–2516. 44, 47, 56, 58, 62

[54] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780. 44, 53

[55] Hong, S. and Lee, M. (2010). Challenges and direction toward secure communication in the scada system. In *Communication Networks and Services Research Conference (CNSR), 2010 Eighth Annual*, pages 381–386. IEEE. 13

[56] IEC (2012). Communication networks and systems for power utility automation—part 90-5: use of iec 61850 to transmit synchrophasor information according to ieee c37.118. 20, 43

[57] IEEE Standards Association (2011a). IEEE Standard for Synchrophasor Data Transfer for Power Systems. *IEEE Std C*, 37. 18, 20

[58] IEEE Standards Association (2011b). IEEE standard for synchrophasor data transfer for power systems. 20

[59] ISO New England (2014). Regional electricity outlook. https://www.iso-ne.com/about/regional-electricity-outlook. 10, 92, 93

[60] Kaufman, C., Perlman, R., and Speciner, M. (2002). Network security: private communication in a public world. 22

[61] Khan, R., McLaughlin, K., Laverty, D., and Sezer, S. (2016). Analysis of ieee c37. 118 and iec 61850-90-5 synchrophasor communication frameworks. In *Power and Energy Society General Meeting (PESGM), 2016*, pages 1–5. IEEE. 20

[62] Kim, S. and Lee, J.-y. (2007). A system architecture for high-speed deep packet inspection in signature-based network intrusion prevention. *Journal of Systems Architecture*, 53(5-6):310–320. 39

[63] Kim, T. T. and Poor, H. V. (2011). Strategic protection against data injection attacks on power grids. *IEEE Transactions on Smart Grid*, 2(2):326–333. 46, 56

[64] Kirschen, D. S. (2002). Power system security. *Power Engineering Journal*, 16(5):241–248. 63

[65] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105. 44

[66] Lagendijk, R. L., Erkin, Z., and Barni, M. (2013). Encrypted signal processing for privacy protection: Conveying the utility of homomorphic encryption and multiparty computation. *IEEE Signal Processing Magazine*, 30(1):82–105. 68

[67] Langner, R. (2011). Stuxnet: Dissecting a cyberwarfare weapon. *IEEE Security & Privacy*, 9(3):49–51. 13

[68] Li, X., Liang, X., Lu, R., Shen, X., Lin, X., and Zhu, H. (2012). Securing smart grid: cyber attacks, countermeasures, and challenges. *IEEE Communications Magazine*, 50(8). 15

[69] Li, Z., Shahidehpour, M., Alabdulwahab, A., and Abusorrah, A. (2016). Bilevel model for analyzing coordinated cyber-physical attacks on power systems. *IEEE Transactions on Smart Grid*, 7(5):2260–2272. 9

[70] Liang, G., Weller, S. R., Zhao, J., Luo, F., and Dong, Z. Y. (2017a). The 2015 ukraine blackout: Implications for false data injection attacks. *IEEE Transactions on Power Systems*, 32(4):3317–3318. 5, 13

[71] Liang, G., Zhao, J., Luo, F., Weller, S. R., and Dong, Z. Y. (2017b). A review of false data injection attacks against modern power systems. *IEEE Transactions on Smart Grid*, 8(4):1630–1638. 13, 25, 46

[72] Liu, C., Wang, X. S., Nayak, K., Huang, Y., and Shi, E. (2015). Oblivm: A programming framework for secure computation. In *2015 IEEE Symposium on Security and Privacy*, pages 359–376. IEEE. 112

[73] Liu, L., Esmalifalak, M., Ding, Q., Emesih, V. A., and Han, Z. (2014). Detecting false data injection attacks on power grid by sparse optimization. *IEEE Transactions on Smart Grid*, 5(2):612–621. 46, 56

[74] Liu, L., Khodaei, A., Yin, W., and Han, Z. (2013). A distribute parallel approach for big data scale optimal power flow with security constraints. In *IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pages 774–778. 67, 73

[75] Liu, Y., Ning, P., and Reiter, M. K. (2011). False data injection attacks against state estimation in electric power grids. *ACM Transactions on Information and System Security (TISSEC)*, 14(1):13. 7, 9, 24, 44, 45, 46, 49, 65

[76] Luo, F., Zhao, J., Dong, Z. Y., Chen, Y., Xu, Y., Zhang, X., and Wong, K. P. (2016). Cloud-based information infrastructure for next-generation power grid: Conception, architecture, and applications. *IEEE Transactions on Smart Grid*, 7(4):1896–1912. 92

[77] Luo, X. (2014). Early experience with cloud computing at iso new england. *Presentation at 2014 PES General Meeting*. 92, 93

[78] Manandhar, K., Cao, X., Hu, F., and Liu, Y. (2014). Detection of faults and attacks including false data injection attack in smart grid using kalman filter. *IEEE transactions on control of network systems*, 1(4):370–379. 46

[79] McClure, S., Scambray, J., and Kurtz, G. (2012). *Hacking Exposed 7: Network Security Secrets & Solutions*. McGraw-Hill Osborne Media. 21

[80] McGrew, D. A. and Viega, J. (2004). The security and performance of the galois/counter mode (gcm) of operation. In *International Conference on Cryptology in India*, pages 343–355. Springer. 41

[81] Mirkovic, J. and Reiher, P. (2004). A taxonomy of ddos attack and ddos defense mechanisms. *ACM SIGCOMM Computer Communication Review*, 34(2):39–53. 7

[82] Mo, Y., Kim, T. H.-J., Brancik, K., Dickinson, D., Lee, H., Perrig, A., and Sinopoli, B. (2012). Cyber–physical security of a smart grid infrastructure. *Proceedings of the IEEE*, 100(1):195–209. 5, 8, 14, 16

[83] Monticelli, A., Pereira, M. V. F., and Granville, S. (1987). Security-constrained optimal power flow with post-contingency corrective rescheduling. *IEEE Transactions on Power Systems*, 2(1):175–180. 63, 64, 72

[84] Morris, T., Pan, S., Lewis, J., Moorhead, J., Reaves, B., Younan, N., King, R., Freund, M., and Madani, V. (2011). Cybersecurity testing of substation phasor measurement units and phasor data concentrators. In *The 7th Annual ACM Cyber Security and Information Intelligence Research Workshop (CSIIRW)*, pages 12–14. 16, 29

[85] Naehrig, M., Lauter, K., and Vaikuntanathan, V. (2011). Can homomorphic encryption be practical? *Proceedings of the 3rd ACM workshop on Cloud computing security workshop - CCSW '11*, page 113. 95, 98

[86] Nagi, J., Yap, K., Tiong, S., Ahmed, S., and Mohammad, A. (2008). Detection of abnormalities and electricity theft using genetic support vector machines. In *TENCON 2008-2008 IEEE Region 10 Conference*, pages 1–6. IEEE. 47

[87] Nagi, J., Yap, K. S., Tiong, S. K., Ahmed, S. K., and Mohamad, M. (2010). Nontechnical loss detection for metered customers in power utility using support vector machines. *IEEE transactions on Power Delivery*, 25(2):1162–1171. 47

[88] Nayak, K., Wang, X. S., Ioannidis, S., Weinsberg, U., Taft, N., and Shi, E. (2015). Graphsc: Parallel secure computation made easy. In *IEEE Symposium on Security and Privacy*, pages 377–394. 68, 91

[89] Nikolaenko, V., Ioannidis, S., Weinsberg, U., Joye, M., Taft, N., and Boneh, D. (2013a). Privacy-preserving matrix factorization. In *Proceedings of the ACM SIGSAC conference on Computer & communications security*, pages 801–812. 68

[90] Nikolaenko, V., Weinsberg, U., Ioannidis, S., Joye, M., Boneh, D., and Taft, N. (2013b). Privacy-preserving ridge regression on hundreds of millions of records. In *Security and Privacy (SP), 2013 IEEE Symposium on*, pages 334–348. 95, 98

[91] NIST Smart Grid (2010). Introduction to nistir 7628 guidelines for smart grid cyber security. *NIST Special Publication*, 154. 43

[92] Nocedal, J. and Wright, S. (2006). *Numerical optimization*. Springer Science & Business Media. 85

[93] Oded, G. (2004). *Foundations of Cryptography. Basic Applications, vol. 2*. Cambridge University Press, New York. 88

[94] Oliver, R. (2001). Countering syn flood denial-of-service attacks. In *Invited Talks of USENIX Security Symposium*. 37

[95] Ozay, M., Esnaola, I., Vural, F. T. Y., Kulkarni, S. R., and Poor, H. V. (2016). Machine learning methods for attack detection in the smart grid. *IEEE Transactions on Neural Networks and Learning Systems*, 27(8):1773–1786. 44, 45, 47, 56, 62

[96] Pai, M., Athay, T., Podmore, R., and Virmani, S. (1989). Energy function analysis for power system stability. *Springer*. 60

[97] Paillier, P. (1999). Public-key cryptosystems based on composite degree residuosity classes. In *Proceedings of the 17th International Conference on Theory and Application of Cryptographic Techniques*, EUROCRYPT'99, pages 223–238, Berlin, Heidelberg. Springer-Verlag. 65, 74, 100, 109

[98] Phan, D. and Kalagnanam, J. (2014). Some efficient optimization methods for solving the security-constrained optimal power flow problem. *IEEE Transactions on Power Systems*, 29(2):863–872. 65, 67

[99] Phan, D. T. and Sun, X. A. (2015). Minimal impact corrective actions in security-constrained optimal power flow via sparsity regularization. *IEEE Transactions on Power Systems*, 30(4):1947–1956. 64, 67

[100] Rahman, M. A. and Mohsenian-Rad, H. (2012). False data injection attacks with incomplete information against smart power grids. In *Global Communications Conference (GLOBECOM), 2012 IEEE*, pages 3153–3158. IEEE. 7

[101] Research, P. (2011). Pike Research Clean Tech Market Intelligence. `http://www.pikeresearch.com/research/smart-grids-in-europe`. Accessed: 2018-03-20. 1

[102] RISI (2011). The repository for industrial security incidents (risi) annual report 2011. Accessed: 2018-03-10. 13

[103] Rosic, D., Novak, U., and Vukmirovic, S. (2013). Role-based access control model supporting regional division in smart grid system. In *Computational Intelligence, Communication Systems and Networks (CICSyN), 2013 Fifth International Conference on*, pages 197–201. IEEE. 38

[104] Rossow, C. (2014). Amplification hell: Revisiting network protocols for ddos abuse. In *In Proceedings of the 2014 Network and Distributed System Security Symposium, NDSS*. Citeseer. 7

[105] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *nature*, 323(6088):533. 52

[106] Sadeghi, A.-R., Schneider, T., and Wehrenberg, I. (2009). Efficient privacy-preserving face recognition. In *International Conference on Information Security and Cryptology*, pages 229–244. Springer. 68

[107] Šandi, S., Krstajić, B., and Popović, T. (2016). pypmu—open source python package for synchrophasor data transfer. In *Telecommunications Forum (TELFOR), 2016 24th*, pages 1–4. IEEE. 40

[108] Schuster, M. and Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681. 53

[109] Stallings, W. (2010). *Network Security Essentials: Applications and Standards*. Prentice Hall Press, Upper Saddle River, NJ, USA, 4th edition. 21

[110] Stallings, W. (2013). *Cryptography and Network Security: Principles and Practice*. Prentice Hall Press, Upper Saddle River, NJ, USA, 6th edition. 8

[111] Stewart, J., Maufer, T., Smith, R., Anderson, C., and Ersonmez, E. (2011). Synchrophasor security practices. In *14th Annual Georgia Tech Fault and Disturbance Analysis Conference*. 38

[112] Sun, C.-C., Hahn, A., and Liu, C.-C. (2018). Cyber security of a power grid: State-of-the-art. *International Journal of Electrical Power & Energy Systems*, 99:45–56. 5, 8, 14, 15, 16

[113] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9. 60

[114] Tavallaee, M., Bagheri, E., Lu, W., and Ghorbani, A. A. (2009). A detailed analysis of the kdd cup 99 data set. In *Computational Intelligence for Security and Defense Applications, 2009. CISDA 2009. IEEE Symposium on*, pages 1–6. IEEE. 56

[115] Tong, Y. (2015). Data security and privacy in smart grid. *PhD diss., University of Tennessee*. 5, 8, 14, 16

[116] Tong, Y., Sun, J., and Sun, K. (2015a). Privacy-preserving spectral estimation in smart grid. In *IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pages 43–48. 67, 83, 95

[117] Tong, Y., Sun, J., Sun, K., and Li, P. (2015b). Outsourcing power system simulations. In *2015 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6. 10, 95, 96

[118] Veugen, T. (2010). Encrypted integer division. In *IEEE International Workshop on Information Forensics and Security*. 84

[119] Veugen, T. (2011). Comparing encrypted data. *Multimedia Signal Processing Group, Delft University of Technology, The Netherlands, and TNO Information and Communication Technology, Delft, The Netherlands, Tech. Rep*. 82

[120] Wang, B. and Sun, K. (2015). Power system differential-algebraic equations. *arXiv preprint arXiv:1512.05185*. 101

[121] Wang, C., Cao, N., and Li, J. (2010). Secure ranked keyword search over encrypted cloud data. *2010 IEEE 30th International Conference on Distributed Computing Systems*, pages 253–262. 98

[122] Wang, C., Ren, K., and Wang, J. (2011). Secure and practical outsourcing of linear programming in cloud computing. In *2011 Proceedings IEEE INFOCOM*, pages 820–828. IEEE. 67, 95

[123] Wang, Q., McCalley, J. D., Zheng, T., and Litvinov, E. (2013). A computational strategy to solve preventive risk-based security-constrained opf. *IEEE Transactions on Power Systems*, 28(2):1666–1675. 67

[124] Xu, W., Trappe, W., Zhang, Y., and Wood, T. (2005). The feasibility of launching and detecting jamming attacks in wireless networks. In *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*, pages 46–57. ACM. 37

[125] Yang, Q., Yang, J., Yu, W., An, D., Zhang, N., and Zhao, W. (2014). On false data-injection attacks against power system state estimation: Modeling and countermeasures. *IEEE Transactions on Parallel and Distributed Systems*, 25(3):717–729. 25, 46

[126] Yao, A. C., Yao, A. C., Yao, A. C., and Yao, A. C. (1982). Protocols for secure computations. In *Foundations of Computer Science, 1982. SFCS '08. 23rd Annual Symposium on*, pages 160–164. 94

[127] Yao, A. C.-C. (1986). How to generate and exchange secrets. In *Foundations of Computer Science, 1986., 27th Annual Symposium on*, pages 162–167. 94

[128] Young, E. . (2014). Cybersecurity: how safe is your smart grid? Accessed: 2018-03-10. 13

[129] Yu, S., Tian, Y., Guo, S., and Wu, D. O. (2014). Can we beat ddos attacks in clouds? *IEEE Transactions on Parallel and Distributed Systems*, 25(9):2245–2254. 7

[130] Yuan, J. and Yu, S. (2013). Privacy Preserving Back-Propagation Neural Network Learning Made Practical with Cloud Computing. *IEEE Transactions on Parallel and Distributed Systems*, 99(1):1–1. 95

[131] Zeigler, B. P., Praehofer, H., and Kim, T. G. (2000). *Theory of modeling and simulation: integrating discrete event and continuous complex dynamic systems*. Academic press. 98

[132] Zhu, S., Setia, S., Jajodia, S., and Ning, P. (2004). An interleaved hop-by-hop authentication scheme for filtering of injected false data in sensor networks. In *Security and privacy, 2004. Proceedings. 2004 IEEE symposium on*, pages 259–271. IEEE. 25

Vita

Xiangyu Niu was born in Jinan, China. He received B.S. degree from Shandong University in 2013. He is pursuing Ph.D. degree in the Department of Electrical Engineering and Computer Science (EECS) at the University of Tennessee, Knoxville and is currently a research assistant under the supervision of Dr. Jinyuan Sun. His research interest includes but not limited to data security and privacy, applied cryptography, machine learning, and cyber-physical security for the smart grid.