Greenbench: A Benchmark for Observing Power Grid Vulnerability Under Data-Centric Threats

Mingkui Wei and Wenye Wang
Department of Electrical and Computer Engineering
North Carolina State University, Raleigh, NC, USA
Email: {mwei2, wwang}@ncsu.edu

Abstract—**Smart grid is a cyber-physical system which integrates communication networks into traditional power grid. This integration, however, makes the power grid susceptible to cyber attacks. One of the most distinguished challenges in studying the aftermath of cyber attacks in smart grid lies in *data-centric* threats. Even though such attacks are critical to the information network, they will result in much more Domino-like impact than they behave in cyber world. This is because for an information-centric network, distorted or delayed information undermines services and applications. But in power grid, these data-centric attacks may result in instable power systems, and further detrimental impact of power supplies. In this paper, we present *Greenbench*, a benchmark that is designed to evaluate real-time power grid dynamics in response to data-centric attacks. The simulation results provide several counter-intuitive suggestions to both smart grid security research and deployment.**

I. Introduction

Smart grid is an emerging cyber-physical system which is expected to replace traditional power grid in near future. Traditional power grid has been running for decades without significant changes on its infrastructure and begins to show its inability as the demand for power delivery and consumption boosts in recent years. One main reason which causes the inefficiency of traditional power grid is the lack of a full-fledged communication infrastructure. Although there exists a control and monitor network which is built above the traditional power grid, most power devices still operate in an isolated manner and their operation is based on electrical properties rather than information exchange. For example, a relay makes the decision to open a circuit breaker only when it detects the current on a feeder exceeds the threshold, it neither tells other relays its own status nor takes information from other relays to help itself make a decision. The lack of information exchange makes traditional power grid fragile because in many situations it is too late to take action when there is a noticeable physical change. No examples are more demonstrative than the 2012 India power outage [1] and the 2003 US-Canada blackout [2], where the initial and minor physical failure was neglected, and the consequent cascading fault is overwhelming and unstoppable.

As a prospective replacement to traditional power grid, smart grid promises a more reliable, effective and efficient power delivery and distribution by integrating advanced communication technologies into traditional power grid. This integration, however, brings a new host of vulnerabilities stem from Internet and opens the door for potential adversaries to tear down a physical system through a cyber attack.

Being aware of the risks, researchers begin to study potential cyber attacks and develop defense schemes to protect this cyber-physical system [3], [4]. However, a practical security solution remains daunting partly because the lack of a commonly recognized platform to evaluate the attack/defense scheme. Question arises when we try to classify various attacks so that we could develop protection solutions in a prioritized way: *How do we analyze, simulate, and evaluate the physical impact caused by a cyber attack in smart grid?*

To address this question, we focus on the *data-centric* threats in smart grid. Even though such attacks are critical to the information network, they will result in much more cascading impact than they behave in cyber world. This is because for an information-centric network, distorted or delayed information undermines services and applications. But in power grid, these data-centric attacks may result in bursty traffic of power flows, instable power systems, and further detrimental impact of power supplies.

A data-centric attack in cyber system aims at gaining advantage by manipulating data exchanged within this system. Although vary in form, the basic attributes of data-centric attacks always lies in one or more of the three categories: Confidentiality, in which the attacker gains access to data which is not supposed to be disclosed to him; Integrity, in which the attacker *distort* the content of data; and Availability, in which the attacker *block* or *delays* the data delivery to legitimate user. These three attributes are the basis of information security and the breach on any of them may cause disastrous consequence.

Critical as they are in the cyber domain, the impact and destructiveness of date-centric attacks could be amplified significantly when being brought into cyber-physical systems like smart grid. From academic researches such as the false data injection attack [5] which points out the design flaw of the monitoring system in modern power grid, to practical attacks like the Stuxnet [6] which destroys nuclear power plant by infecting and distorting control data, it is obvious that data-centric cyber attacks is real and the demand for the defence is urgent.

In this paper, we present *Greenbench*, a benchmark that is developed to capture power system dynamics, such as real-time readings and variations of current and voltage. To evaluate

This work is supported by NSF-ERC FREEDM.

the impact of data-centric threats, we carry out three case studies, *delayed price information* for Advanced Metering Infrastructure (AMI), *modified load information* for load distribution and dispatch, and *composite data* for energy management. The remainder of this paper is organized as follows. In Section II we describe the framework of *Greenbench*, details, challenges and our solutions. In Section III, we discuss the impact of data-centric attacks in power grid by using *Greenbench*. And in Section IV we conclude our work.

II. *Greenbench*: System Framework and Design

In this section, we describe the background in developing cross-domain tools for smart grid, and the framework of *Greenbench* by starting from its underlying power systems, *Green Hub*. Then we present two main challenges for system implementation in detail, that is, synchronization between discrete event and continuous simulators, and data exchange during simulation.

A. Background

The cyber security issues in smart grid have received tremendous attentions in the past few years. In particular, many research efforts have simulation-based approaches, mainly because this method would provide a good understanding of such a complex system without interrupting power systems in operation. For instance, a cyber-physical testbed for smart grid protocol is developed by using DETER [7], which is a well designed cyber-physical framework. In this study, a case study regarding AMI was demonstrated [8], focusing on how to parse communication protocol used by smart meters, without showing the results due to such attacks. The impact of DoS attack in AMI is also studied in ns2 [9], which showed the distorted meter readings.

Through modeling physical device (e.g., the dynamical system model for generator)[10], the physical impact due to cyber attacks is quantified by using a graph-based approach without having interaction with the cyber-domain. Moreover, the Electric Power and Communication Synchronizing Simulator (EPOCHS) introduced in [11] is a study focusing on the power grid study, where communication was treated as a way to pass data.

Our proposed framework, *Greenbench*, makes new contributions in two-fold. First of all, *Greenbench* is a cross-domain simulation platform that includes an underlying power system *overlayed* by a communication system. In such a way, we are able to capture the impact of cyber attacks in power systems in real-time, unlike networking simulations as [9] or physical systems like [10], [12], [11]. Second, we aim to study the consequences of *data-centric attacks* rather than manipulation of communication protocols like [7]. The benefits of our efforts is that there might be many attacks or manipulation schemes to attack smart grid, however, the resulting of attacks and countermeasures for data integrity will be revealed in ultimate data received, which could be *delayed* and/or *distorted*. To this end, our study is able to demonstrate the direct impact of security attacks at the power system level, either due to compromised smart meters in AMI or DoS attacks to messages in transmission.

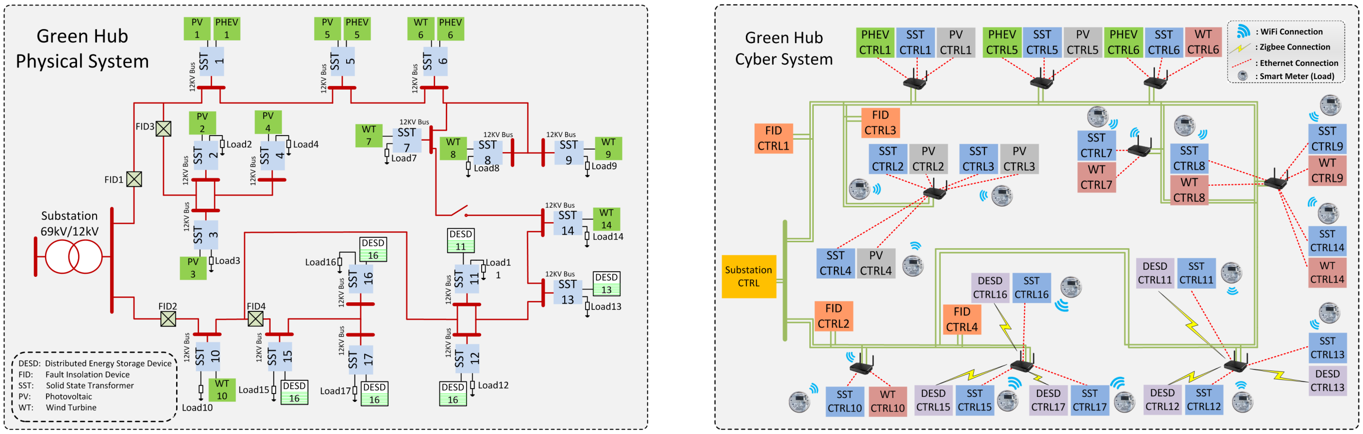
B. Green Hub: A Micro Smart Grid

Our objective is to develop a *cross-domain* simulation platform that can be used to demonstrate the interaction and interdependency of cyber attacks and power grid in real-time. As a platform, we use *Green Hub* as the underlying physical system for our study.

The Green Hub system is a novel distribution level micro-grid which has been developed at the Future Renewable Electric Energy Delivery and Management (FREEDM) systems center for the study of power management strategies [13]. The Green Hub is abstracted from an actual residential distribution system, which is a 230kV/22.86kV substation along with two 22.86kV distribution feeders in the Raleigh area, while the substation voltage is reduced to 69kV/12kV to fit our study purpose. The Green Hub contains various innovative power devices developed in FREEDM center, such as the Solid State Transformer (SST), and the Fault Isolation Devices (FIDs), and it is also connected to green energy sources such as the Photovoltaic (PV) and Wind Turbine (WT). All those devices are equipped with Intelligent Electronic Devices (IEDs), which are ARM-based embedded systems used for real time control/monitor and communication. Those IEDs interact with each other to make the Green Hub a self-autonomous micro smart grid which could either be connected to main power grid or operate in an isolated mode.

In order to use this power subsystem for our study, we have to deal with two issues as follows:

- **System abstraction:** An actual system includes a large number of various devices which makes it improper for study and simulation, thus it is necessary to simplify and abstract a high-level system with a suitable size and omit the minor details. The abstracted power system is shown in Fig. 1(a), which is a 17-bus power distribution system. Each bus is connected with a SST, which is able to implement bi-directional energy flow and DC/AC transformation. Each SST is connected with a load (Load represents AC load, PHEV represents DC load), and a renewable energy source (PV, WT, or DESD). To ensure the reliability of the system, four FIDs are deployed on different feeder segments, which will open the circuit breaker and isolate failure from upper level power grid in case of a fault happens.
- **Domain mapping:** The challenge here is to map the physical domain into cyber domain by replacing each physical devices with its corresponding IEDs, the mapped cyber domain system is shown in Fig. 1(b). Smart meter is used to represent AC load as it is the typical controller for AC load such as households or buildings. Also shown in this figure is the different network access methods for various IEDs (controllers), which reflect the enabling works undergoing in FREEDM center. Specifically, the SST, PHEV, PV and WT controllers are connected to the communication network using Ethernet, the DESD



(a) Green Hub Physical System

(b) Green Hub Cyber System

Fig. 1. Cyber-physical system.

controller is connected using Zigbee, and the smart meter uses wireless to access the network.

C. Greenbench *Framework and Implementation*

The framework of *Greenbench* with its software implementation architecture is shown in Fig. 2. We briefly introduce the architecture and the functionality of each block of the framework, while leave detailed description and design challenges to next section.

The *Greenbench* framework is functionally composed by two parts (simulators), the physical part (PSCAD) and the cyber part (OMNeT++). The physical and cyber domain model shown in Fig. 1 is built in their corresponding part, and the two parts interact through two interfaces, the *interactor*, and the *buffer files*.

i. Interactor: The interactor is a special application build *within* OMNeT++. It handles synchronization between the two simulators by switching simulation sequence between them.
ii. Buffer file: Buffer file is a pool of binary files used by the two domain simulation to exchange data in real time.

The physical domain part could be further split into 3 function blocks.

i. Interface: The interface function block provides Human-Machine interaction. The simulation control such as begin, pause, and stop formulate the basic functionality of this block. Within the interface block, user could build the system model, as well as observe the system performance graphically.
ii. EMTDC simulation engine: EMTDC is a electro-magnetic transients simulation engine which takes device parameters as its input, computes system state by solving differential equations, and exports the result as output.
iii. C interface: The C interface is a bi-directional interface written with C language. It fetches data from buffer files and export it to EMTDC; and receives the result from EMTDC, and write the result value into buffer files.

The cyber domain part includes 2 function blocks:

i. OMNeT++ simulator/INET framework: The OMNeT++ is a platform which provides basic graphic interface and simulation control (begin, stop, etc). The INET is a framework built above OMNeT++ which provides Internet-specific support, such as wireless/wired channels, and TCP/UDP hosts. The cyber domain entities (IEDs), as well as the network topology are built in OMNeT++ simulator using models provided by INET framework.
ii. C++ interface: Same as the C interface in physical domain part, the C++ interface is used to import/export data from/to buffer files except it is written by C++.

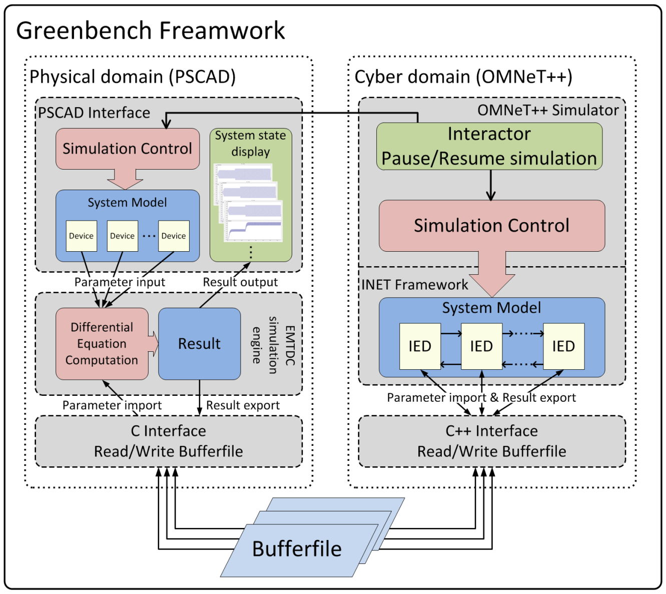
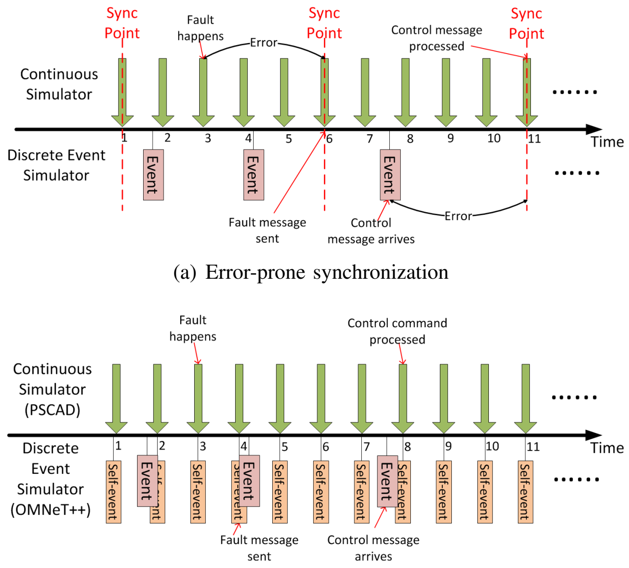


Fig. 2. Software implementation of *Greenbench*.

D. Design Challenges

As we described earlier, *Greenbench* is essentially a power grid with overlayed communication platform. Conceptually, this seems intuitive and trivial. However, the implementation of such a cross-domain system is not easy. In spite of the detailed procedures of implementing functions of each power device and access communication techniques and protocols, we have to deal with two high-level challenges, *synchronization and data exchange.*

D.1) Synchronization of Continuous and Discrete Events: Most of the network simulators, such as ns2, ns3, OPNET, OMNeT++, etc., are *discrete event* simulators. A discrete event simulator is driven by queued events, each event occurs at a particular time and marks a change of system state. Between two consecutive events, it is assumed that the system state will remain unchanged and nothing interested happens. Therefore the simulation can directly jump over time from one event to next event. On the contrary, power system simulators, such as PSCAD and RTDS [14] are *continuous* simulators, which solves differential equations at a fixed time step. The simulation of a continuous simulator is executed step by step without any time step could be passed over.



(a) Error-prone synchronization

(b) Error-free synchronization

Fig. 3. Synchronization of continuous and discrete events.

The different ways to handle simulation time causes the synchronization problem. As shown in Fig. 3(a), it is possible that during two events of OMNeT++, which will be used for *Greenbench* to setup communication platforms, a power system simulator may generate several messages at different time points. These messages pile in the buffer of a network simulator and wait for the next event. However, when the next event happens, all messages will be processed and stamped with the same time point, which makes them unsynchronized.

To tackle similar issues, in [11], the authors set a synchronization point, and use an agent to periodically check messages from both simulator at each point. However this mechanism may cause cumulative errors [12], and the use of an external agent will cause unnecessary complexities. For instance, as shown in Fig. 3(a), at time 3 a fault happens in power system, but the simulator has to store this fault state until next sync point which is at time 6; on the other hand, between time 7 and 8, there is a control message arrives from control center, this message can not be processed either until time 11.

To implement a perfect synchronization while minimize the cost, we adopt a simpler yet effective mechanism to handle the interaction by fully utilizing the flexibility of OMNeT++.

OMNeT++ is an event driven simulator, and the event is based on messages exchanged between modules in the simulated system. For instance, an event is created when a "TCP application" module tries to pass a message to a "transmission channel" module. Among various message types, there is one type of message called self-message. The self-message is send by a module to itself at a scheduled time in the future, which is mainly used as an internal initiator to keep a module "alive". For example, to build a model in which the host sends one TCP packet every 10 seconds, we could set the self-message to be sent with a 10 seconds interval, and on receiving the self-message the the host will send the TCP packet. Without the self-message, a module will only passively response to a message received from external.

The existence of the self-message enables the OMNeT++ to act as a continuous simulator which could be perfectly synchronized with PSCAD. Particularly, we design a special application within OMNeT++ model, which we call the "interactor". The interactor is not part of the communication network. It handles the interaction of the two simulators by sending self-message with a fixed time interval, and switching simulation operation between them. Particularly, at the time when the self-message is received, the interactor halts OMNeT++ execution, and passes the execution to PSCAD to simulate the next time step; when PSCAD completes the next step, it halts the PSCAD execution and resumes OMNeT++ execution until next self-message arrives. The error-free synchronization is shown in Fig. 3(b), here we assume at each simulation time step, the OMNeT++ executes first. As shown in Fig. 3(b), the fault happens at time 3 in power system, and the fault data is exported to OMNeT++ immediately. As at this time the OMNeT++ has completed this time step, the simulation proceeds to time 4. At time 4, the self-event happens at OMNeT++, in which the IED detects the imported data, and send it out within this simulation step. As another example, between time 7 and 8, the control message arrives at the IED, and this message is exported to PSCAD at time 8 by the self-event. Right after that, the PSCAD gains execution and the power device imports and executes the control message.

The self-message time interval is designed to be adjustable which allows user to choose different simulation accuracy both in PSCAD and in OMNeT++.

The interactor also has a definable start and stop time, with which user is able to specify the time span during which the two simulator interact. Consider again in Fig. 3(b). It is obvious that although the simulation was executed for 11 time units, the actual time during which the two simulator interact is between time 3 and time 4, and time 7 and time 8. A definable start and stop time allow us to set two simulator begin to interact before time 3/7 and stop after time 4/8, which could save a significant amount of simulation time as the interaction costs much longer time than any single simulation.

D.2) Data exchange: Another key factor to implement the integration is data exchange mechanism between the two simulators. The physical devices in PSCAD will generate status information during simulation, and this information needs to be passed to OMNeT++ and sent via Internet in the form of TCP/UDP packet; while control center in OMNeT++ will send control commands to device controllers via Internet, and those commands have to be directed to physical devices. An efficient real-time data exchange between the two simulators is needed to implement the cyber-physical simulation platform.

In [11], [12] a Runtime Infrastructure or a Globe Scheduler process is used as a globe manager to handle the interaction, this implementation is effective but lacks efficiency. Compared to an external globe manager, a better choice is to use Inter-Process Communication (IPC) technique to directly exchange data between the two processes. IPC is a technique used to make several stand-alone processes communicate with each other. The commonly used implementations of IPC are named pipes (fifo), Windows socket, and shared memory, etc, and the basic idea of which is to have one process store the desired data at one place (fifo files, sockets, or a segment of memory), and have another process fetch it at the same place.

We implement the data exchange in the *Greenbench* in a simpler way: for each cyber-physical device pair, we create 2 binary files, which we call "buffer files". Those files are "directional", one of which are used by physical device to export data to cyber device, such as a reading from a smart meter. And another file is used for cyber device to pass data to physical device such as a control command. For each simulation step, the two simulators read buffer files, import data if exists, execute simulation for one time step, and export data to buffer files if needed.

III. Delayed and Distorted Data-Centric Attacks

In the data-centric attack, the attacker aims at gaining advantage or cause damage by manipulate the data exchanged between network entities. This data-centric attack is even more dangerous in smart grid because instead of interrupt applications and services in cyber world, it will disturb and damage the critical infrastructure, and potentially cause disastrous loss which is not confined only in terms of economic. In order to better understand its impact, find effective solutions as well as instructive suggestions, we hereby study the data-centric attack in smart grid by focusing on smart meter targeted attacks.

Smart meter in AMI is one of the most vulnerable components in smart grid. For the first reason, it is physically accessible to public; For the second , it uses wireless communication which is susceptible to jamming attack [15] and easy to be overheard [16]; For the last and most important, it is usually overlooked by manufactures and was not designed to resist any cyber attack [8], [17]. However, a system is as strong as its weakest link, and it remains an open question that *whether the omitted security feature on smart meter is reasonable*. To address this question, we select three cases from different security aspect and study them in *Greenbench*, which include a delayed data attack, a distorted data attack, and a composite attack.

The metrics usually used to observe the state of a power system is voltage, current, real power and reactive power. For the simulated power system, the voltage on each point will remain unchanged unless an overload happens, nut the current keeps changing with variation of load; while the trend for real power and reactive power change follow the same pattern during our simulation. Therefore, we use current and real power to illustrate the state change of the Green Hub hereafter.

For easy description, we divide the Green Hub shown in Fig. 1(a) into 4 sections: Section 1 starts after FID1 and includes load 1, 5, 6, 7, 8, and 9; Section 2 starts after load 10 and includes load 11, 12, 13, and 14; Section 3 starts after FID3 and includes load 2, 3, and 4; And section 4 starts after FID4 and includes load 15, 16, and 17. Note that load 10 does not belong to either sections.

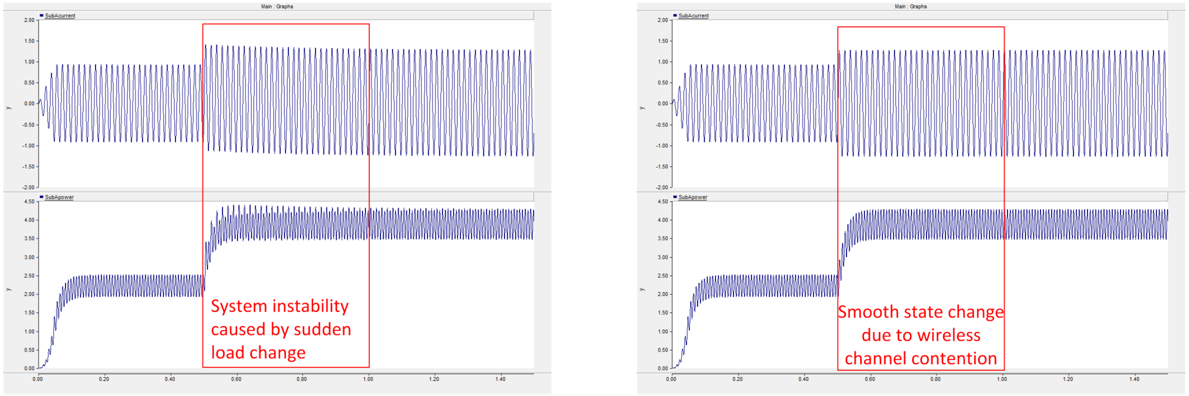
A. Delayed Price Information in AMI

In this case study we simulate and analyze the "jamming the price signal attack" which was proposed in [15]. Particularly, it is assumed that the power consumption at consumers is based on the pricing information, which is a continuously changed variable. The pricing information is sent to consumers (smart meters) by an aggregator via wireless link and the attacker is able to jam the pricing signal within a certain area. During the jamming, the consumers will remain the power consumption amount because they do not have the up-to-date pricing information. When there is a significant change of the pricing information, the attacker stops jamming. The sudden change of the pricing information will cause a significant change on power consumption in a short time, and consequently affects the power grid stability.

In this case we assume that the attacker compromised the load controller (smart meter) 11, 12, 13, and 15, 16, 17, which locate within a nearby area geographically. We also assume the extreme case that during the jamming attack, consumers simply do not consume any power, and then operate under full load when the jamming stops and updated pricing signal is received. As a comparison, we also analyze this scenario and simulate it in single domain using PSCAD.

Single domain simulation*:* If being considered only in the power system domain, this attack could be modeled with a simple scenario, in which a sudden load change happens at a certain time point. The single domain simulation result is shown in Fig. 5(a), in which the current and real power change at the substation transformer are depicted.

As shown in Fig. 5(a), the jamming stops at 0.5 second. Because the sudden increase of large amount of load, the substation transformer undergoes a current hike which is higher than normal current, and the current converges to normal value after 1 second (at 1.5 second). The real power output at the substation transformer also shows instability for 0.5 second and returns to normal after 1 second.



(a) Substation current change (single domain)

(b) Substation current change (*Greenbench*)

Fig. 5. Jamming the price signal attack simulation.

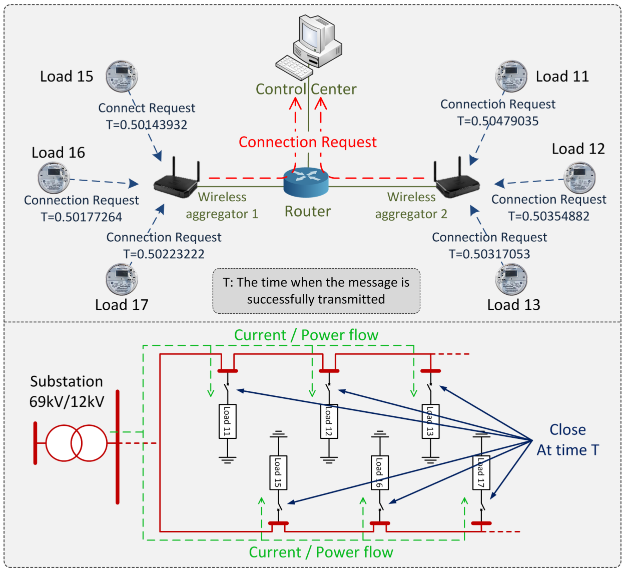


Fig. 4. Jamming the Price Signal attack.

Greenbench simulation*:* When being considered in cyber-physical cross domain, however, the single domain scenario setup is over-idealistic. In practice the smart meters won't be able to communicate with the wireless aggregator exactly at the same time, because wireless channel can only be used by one host at any time. A more realistic simulation is deployed in the *Greenbench*, and the simulation setup is shown in Fig. 4. Wireless aggregator 1 (WA1) is the access point for load 15, 16, and 17, while wireless aggregator 2 (WA2) is the access point for load 11, 12, and 13. There is no interference between WA1 and WA2 area, but hosts within each area will contend to access the wireless channel. And the physical load is assumed to be connected to the power system immediately when its load controller gains the access to its WA and its connection request is received by the control center.

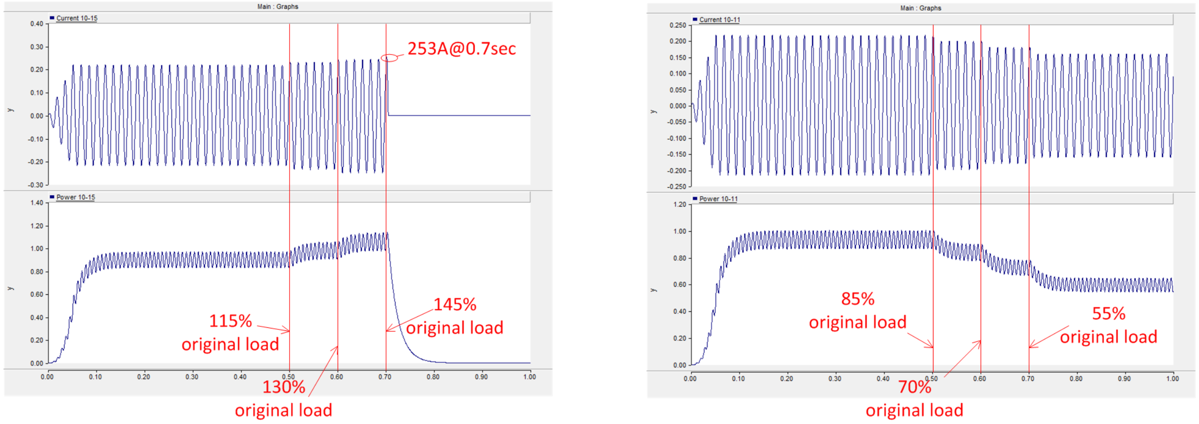
The *Greenbench* simulation result is shown in Fig. 5(b), and the difference from Fig. 5(a) is obvious. Because of wireless channel contention, the connection requests from those 6 loads do not arrive at control center at the same time, and hence the physical loads also take turns to be connected to main power grid. Although the time between each load get connected is very short, it is enough for the power grid to be prepared for the load change, and therefore the current and real power change is much more smooth than in Fig. 5(a), which indicates the system stability is unlikely to be impacted.

Remark 1: In this case, the attack causes a real load change, and the attacker's goal is to cause an instability to power system by the sudden load change. A similar attack named "distributed internet-based load altering attack" [18] also follows this type, in which the attacker is assumed gained the control of smart meters over a large area, and by turning off a large amount of household load, e.g., water heater in 1000 homes, the power grid stability is negatively impacted. However, as shown by *Greenbench* simulation, this type of attack actually bears low risk mainly because the contention period of wireless communication acts as a buffer which mitigates the "sudden" change so that the power grid has enough time to prepare for the load change.

B. Distorted Load Attacks

In this case the Load Redistribution (LR) attack [19] is simulated. The LR attack is a special type of the false data injection attack [5]. The false data injection attack refers to an attack in which carefully designed false data could be added to certain group of monitored data in power grid, however, those false data can not be detected by the *state estimation* algorithm which is used to detect bad data in power grid. This false data is accepted and used by the control center to make decision, although it is not in consistent with real device status, and this inconsistency may cause unpredictable damage to power grid.

In the LR attack, the author put some constrains on the attackable nodes in smart grid, which makes LR attack more practical and easier to be launched. Particularly, while in original false data injection attack the author treat each node homogeneously, in LR attack it is assumed that only the load nodes are attackable. Note in this attack, the attacker's goal is not to change the real load – the power consumed by a device,



(a) Current and power flow through Metet_10_15

(b) Current and power flow through Meter_10_11

Fig. 7. Load redistribution attack simulation in *Greenbench*.

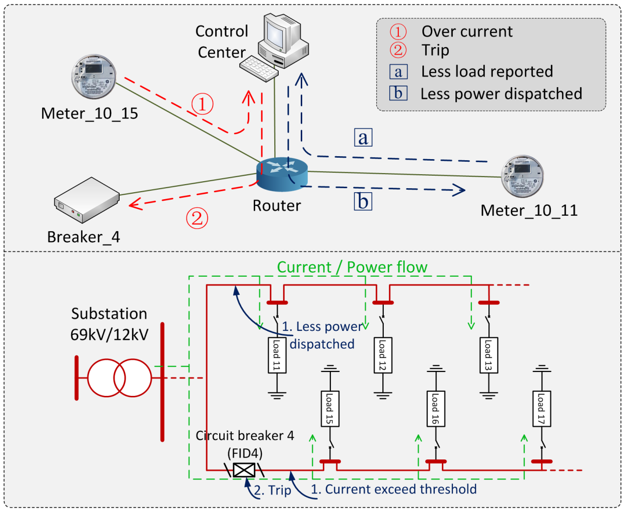


Fig. 6. Load Redistribution attack.

but to modify the load reading, which is the monitored value sent to the control center. And we use *real load* and *load reading* to different the two concepts hereafter.

Same as in case I, we also assume the attacker compromised meters which provide readings for load 11, 12, 13, 15, 16, and 17. Two special constrains of the LR attack are that the overall real load consumption of the attacked area remains the same, while the load reading changes for each specific load does not exceed 50% of its original load. According to these constrains, we setup the attack scenario as following:

1) Assume the attacker increases the load reading at load 15, 16, and 17; and decreases it at load 11, 12, and 13. The total increased load at load 15, 16, 17 and total decreased load at load 11, 12, 13 sum to zero.
2) The attack is launched in 3 time step with 0.1 second time-interval between each step. For each step, at load 15, 16, and 17, the attacker increases their load reading by 15% of their original load, and at the same time he decreases the same amount of load reading at load 11, 12, and 13. The total load reading change for each load is 45% of its original load.
3) Note that in this simulation, our goal is different from [19]. In [19], the goal of the attack is to find a combination of load redistribution which causes the maximum cost, while our goal is to deploy this attack in a real cyber-physical system and study its potential physical impacts rather than its economic cost. Therefore it is unnecessary to solve the optimization problem used in [19].

The simulation setup is shown in Fig. 6. In Fig. 6, the Meter_10_15 and Meter_10_11 are meters which monitor the current and power flow on the feeder segment between load 10 - load 15, and load 10 - load 11, and their sample frequency are set to be 10 samples (messages) per second. The maximum threshold on feeders in both section 2 and section 4 is set to be 250A. And in this simulation we collect only the meter reading from Meter_10_15 and Meter_10_11 as it is intuitive that the feeders in segment 10 - 11 and 10 - 15 hold the maximum current and real power value in their own branches, and thus they are the first ones to fail if there is an over-current on these branches. The Breaker_4 represents the circuit breaker controller of FID 4.

The simulation result is shown in Fig. 7, and the attack steps are described as below:

1) **t=0.5s:** Attacker launches attack. Both branches operate normally and the current remains at 210A.
2) **t=0.5s-0.7s:** Load reading in section 4 increases with 15% per 0.1 sec, while load reading in section 2 decreases with the same pace.
3) **t=0.7s:** Current at section 4 exceeds threshold by reaching 253A, and over-current message is sent to control center. Control center sends trip message to breaker 4, and section 4 loses power.

On the other hand, as shown in Fig. 7(b), because the monitored load decreases in section 2, less power is dispatched to this branch, and consequently the current is much lower than it should be, which will also cause abnormal behavior of power devices in this section.

Remark 2: In this case, the attack dose not change any real load consumption, on the contrary, it modifies the messages sent by meters and aims at confuse the control center of monitored load consumption and real load consumption. As shown by the result, this type of attack is more dangerous. Because the control center is bewildered of the real state, it makes an incorrect decision, which is more harmful than merely a sudden load change.

Remark 3: Greenbench simulation of the two cases suggests a smart grid security solution which is instructive for smart grid security research: *relatively, a single or a set of smart meters being compromised and gained control does not put smart grid under a high risk; as long as the attacker is unable to forge an authentic message, the whole smart grid is safe*. Therefore, compared to fortify smart meter and keep it from being compromised, the we should pay more attention on designing security policies to authenticate messages and detect a bad or inconsistent message even if a meter is compromised.

C. Composite Attacks: Distorted Data and Man-in-the-Middle Attack

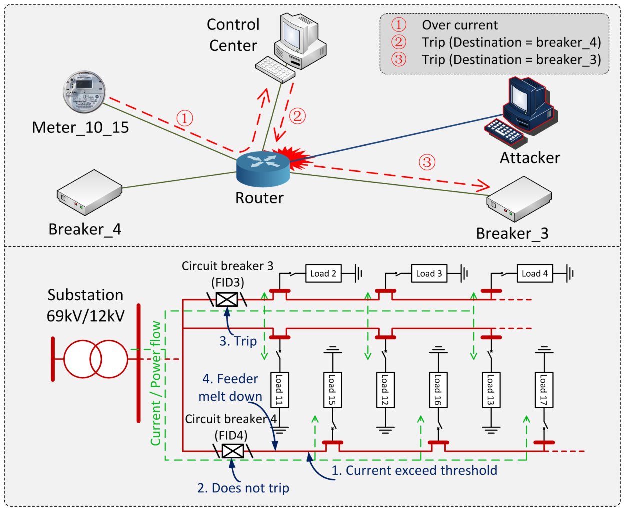


Fig. 8. LR attack and Man-in-the-middle attack.

The power grid is a critical infrastructure and is state-owned in many countries, thus those who sabotage power grid assumes serious crime. It is reasonable to assume the power grid targeted attack is made by clear purpose and therefore the attacker will explore every possibility to maximize the damage. Rather than a single attack, the attacker is highly likely launch multiple attacks which affects more devices.

In this case we assume a skilled attacker combines more than one attacks and tries to cause a more severe impact to smart grid. Specifically, we assume that at the same time the LR attack is launched, the attacker also compromises a router and applies a Man-in-the-middle attack, in which he eavesdrops messages processed by the router, locates the "trip" message send from control center to breaker 4, and modifies the destination address of the "trip" message from breaker 4 to breaker 3. This scenario is shown as in Fig. 8.

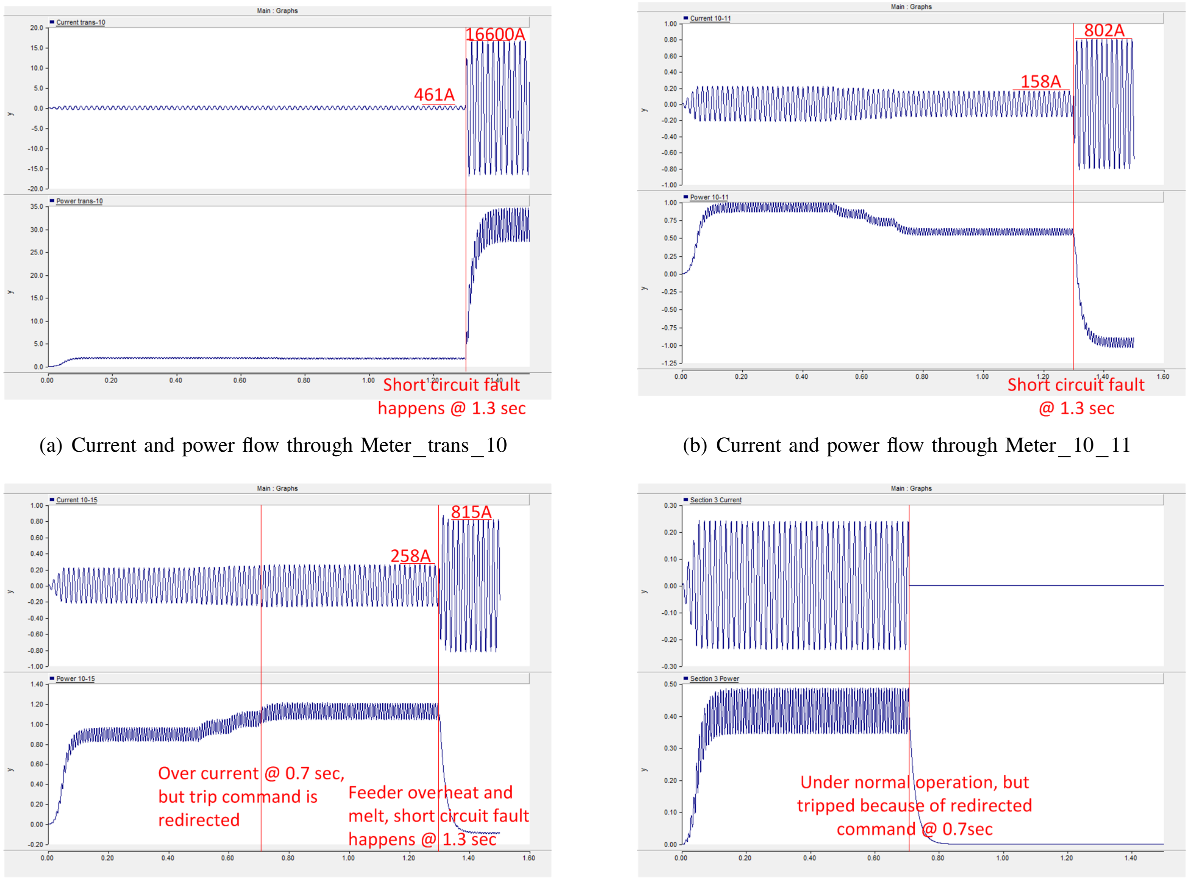
Fig. 9 shows the current and real power value at different points in the Green Hub. In which the "trans_10" denotes the feeder segment between substation transformer and load 10.

Same as in previous case, the LR attack begins at 0.5 second, and at 0.7 second, the monitored current at Meter_10_15 exceeds 250A, and the control center sends the "trip" message to breaker 4. However, because the attacker also compromised the router, the "trip" message sent by control center was modified, and the message destination was changed to breaker 3. As a direct result of the redirected message, breaker 3 trips and causes a blackout of the whole section 3, as shown in Fig. 9(d). On the other hand, because breaker 4 does not receive the "trip" message from control center, the circuit breaker remains closed, which makes the feeder in section 4 run under a over-current situation. At time 1.3 second, 0.5 seconds after running under abnormal condition, the extra heat caused by the over-current causes the feeder to melt and a feeder-to-ground short circuit fault happens, which causes a disastrous impact to the whole power grid, as shown in Fig. 9(a), Fig. 9(b), and Fig. 9(c). Fig. 9(b) and Fig. 9(c) show the current and power flow at section 2 and section 4, in which current jumped almost 4 times of their normal situation; and the power on both branch suddenly dropped to negative, which indicates a reverse current flow. A more severer damage is caused on the feeder segment from substation transformer to load 10, which is shown in Fig. 9(a). The current on this feeder surged from around 450A to 16,600A, more than 30 times of its normal operate value. Such a huge serge will surely cause severe damage to connected power devices, and the transformers and even the substation are also very likely to be damaged, which may serve as a start point of a larger-area cascading failure.

Remark 4: As indicated in the simulation result, the composite attacks cause much severer impact than any of the single attack. This result indicate another *non-intuitive* solution: *although an attack on any single device is unavoidable, its impact could be limited by making it difficult for the attacker to combine various attacks.* The most intuitive solution (yet always being neglected in practice) is to use different login/password for different devices. For more sophisticated solutions, one could deploy a hierarchical security policy, in which different levels of devices are protected by different security methods (physically locked and deploying surveillance camera, using encryption algorithms such as AES, etc).

IV. Conclusions and Future Work

In this paper, we present *Greenbench*, a cross-domain simulation platform which could capture the impact of cyber attacks in power systems. Along with *Greenbench*, We study the data-centric attacks which target at damaging power grid by manipulating the data exchanged between devices. The simulation results convey non-intuitive indications and instructive suggestions to both smart grid security research and deployment. Nonetheless, *Greenbench's* capability is not confined to this, its flexibility and extensibility allows us to analysis and evaluate various smart grid attacks, which is one of our future works. As another future work, we will integrate



(a) Current and power flow through Meter_trans_10

(b) Current and power flow through Meter_10_11

(c) Current and power flow through Meter_10_15

(d) Current and power flow through Meter of section 3

Fig. 9. Attack combination simulation in *Greenbench*.

power grid dedicated communication protocols such as DNP3 and IEC-61850, and evaluate attacks targeting those protocols.

References

[1] I. Consulting, "The economic cost of the blackout: an issue paper on the northeastern blackout, august 14, 2003," *Fairfax, VA: Accessed at: http://www. solarstorms. org/ICFBlackout2003. pdf*, 2003.

[2] "U.S. - canada power system outage task force: Final report on the implementation of task force recommendations," energy.gov/oe/downloads/us-canada-power-system-outage-task-force-final-report-implementation-task-force.

[3] L. Pietre-Cambacedes, C. Chalhoub, and F. Cleveland, "Iec tc57 wg15–cyber security standards for the power systems," *CIGR É Study Committee D*, vol. 2, 2007.

[4] D. Watts, "Security and vulnerability in electric power systems," in *35th North American power symposium*, vol. 2, 2003, pp. 559–566.

[5] Y. Liu, P. Ning, and M. K. Reiter, "False data injection attacks against state estimation in electric power grids," in *16th ACM Conference on Computer and Communication Security*, NY, USA, 2009, pp. 21–32.

[6] "W32.Stuxnet Dossier," www.symantec.com.

[7] J. Mirkovic, T. V. Benzel, T. Faber, R. Braden, J. T. Wroclawski, and S. Schwab, "The deter project: Advancing the science of cyber security experimentation and test," in *Technologies for Homeland Security (HST), 2010 IEEE International Conference on*. IEEE, 2010, pp. 1–7.

[8] T. Yardley, R. Berthier, D. Nicol, and W. H. Sanders, "Smart grid protocol testing through cyber-physical testbeds," in *Innovative Smart Grid Technologies (ISGT), 2013 IEEE PES*. IEEE, 2013, pp. 1–6.

[9] A. AlMajali, A. Viswanathan, and C. Neuman, "Analyzing resiliency of the smart grid communication architectures under cyber attack," in *5th Workshop on Cyber Security Experimentation and Test*, 2012.

[10] D. Kundur, X. Feng, S. Mashayekh, S. Liu, T. Zourntos, and K. Butler-Purry, "Towards modelling the impact of cyber attacks on a smart grid," *Intl Journal of Security and Networks*, vol. 6, no. 1, pp. 2–13, 2011.

[11] K. Hopkinson, X. Wang, R. Giovanini, J. Thorp, K. Birman, and D. Coury, "Epochs: a platform for agent-based electric power and communication simulation built from commercial off-the-shelf components," *Power Systems, IEEE Transactions on*, vol. 21, no. 2, pp. 548–558, 2006.

[12] H. Lin, S. Sambamoorthy, S. Shukla, J. Thorp, and L. Mili, "Power system and communication network co-simulation for smart grid applications," in *Innovative Smart Grid Technologies (ISGT), 2011 IEEE PES*. IEEE, 2011, pp. 1–6.

[13] H. Hooshyar, "System protection for high pv-penetrated residential distribution systems (green hubs)." 2011.

[14] "RTDS," http://www.rtds.com/index/index.html.

[15] H. Li and Z. Han, "Manipulating the electricity power market via jamming the price signaling in smart grid," in *GLOBECOM Workshops (GC Wkshps), 2011 IEEE*. IEEE, 2011, pp. 1168–1172.

[16] C. Efthymiou and G. Kalogridis, "Smart grid privacy via anonymization of smart metering data," in *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*, pp. 238–243.

[17] S. McLaughlin, D. Podkuiko, and P. McDaniel, "Energy theft in the advanced metering infrastructure," in *Critical Information Infrastructures Security*. Springer, 2010, pp. 176–187.

[18] A.-H. Mohsenian-Rad and A. Leon-Garcia, "Distributed internet-based load altering attacks against smart power grids," *Smart Grid, IEEE Transactions on*, vol. 2, no. 4, pp. 667–674, 2011.

[19] Y. Yuan, Z. Li, and K. Ren, "Modeling load redistribution attacks in power systems," *Smart Grid, IEEE Transactions on*, vol. 2, no. 2, pp. 382–390, 2011.