Leveraging Data-Centric Edge Computing to Defend IoT-based Attacks in Power Grids

Bibek Shrestha, Hui Lin
Computer Science and Engineering Department
University of Nevada, Reno
bibek.shrestha@nevada.unr.edu, hlin2@unr.edu

***Abstract*—Internet-of-things (IoT) introduce new attack surfaces for power grids with the usage of Wi-Fi enabled high wattage appliances. Adversaries can use IoT networks as a foothold to significantly change load demands and cause physical disruptions in power systems. This new IoT-based attack makes current security mechanisms, focusing on either power systems or IoT clouds, ineffective. To defend the attack, we propose to use a data-centric edge computing infrastructure to host defense mechanisms in IoT clouds by integrating physical states in decentralized regions of a power grid. By enforcing security policies on IoT devices, we can significantly limit the range of malicious activities, reducing the impact of IoT-based attacks. To fully understand the impact of data-centric edge computing on IoT clouds and power systems, we developed a cyber-physical testbed simulating six different power grids. Our preliminary results show that performance overhead is negligible, with less than 5% on average.**

***Index Terms*—IoT, Edge computing, power grids security**

I. Introduction

Power grid systems integrate information technology (IT) and operational technology (OT) components to perform optimal energy management [1]. IT components are connected by an IP-based communication network, which is isolated from the public Internet; OT components are connected by electrical equipment, such as transmission lines. Utility operators use these IT/OT networks to monitor and control state of physical components, e.g., load demands adjusted by individual users.

However, load units equipped with Wi-Fi access become Internet-of-things (IoTs) and are exposed to an open IoT cloud that is beyond the control of utility operators. Through this IoT cloud, users can employ various mobile applications or web-interface terminals to adjust load demands.

As shown in Figure 1, IoT clouds introduce a new attack surface to power grids with increasing usage of Wi-Fi enabled high wattage appliances, such as water heaters and ovens. Recent studies have revealed that device vulnerabilities enable adversaries to control a large number of IoT devices permeating over a wide area. For example, in Mirai attack, adversaries compromised more than 6,000,0000 poorly-configured Internet-connected cameras to launch the largest-scale Distributed Denial of Service (DDoS) attack. Once applied to the IoT-enhanced power grids, adversaries can significantly change load demands instantly, causing system-wide instability and disruptions [2].

The IoT-based attacks make current security mechanisms, focusing on either power systems or IoT clouds alone, ineffective (see Section II). Closing those research gaps requires a defense in IoT clouds that can integrate physical knowledge of power systems. The unique requirement leads us to edge computing. Edge computing is originally used to increase network performance of large content providers [3], by deploying edge servers in decentralized regions to handle requests from end devices without involving a centralized cloud server (see Figure 1). Each edge server constructs a small cloud environment, often referred to as an edge cloud, in close proximity to end devices to balance communication traffic and serve their requests with short latency [4]. Instead of improving network performance, we propose to *use edge computing to enhance current perimeter-based defenses in IoT clouds*, by collecting physical data from IoT devices in different regions of power systems and using them to determine and enforce security and safety policies.

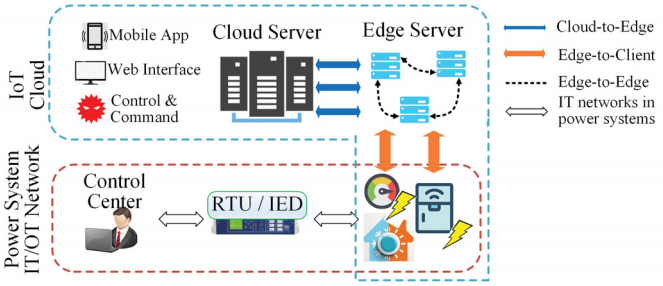


Fig. 1. **Exposing load units to IoT clouds introduces a new attack surface to power systems:** adversaries compromise mobile applications, web interfaces, or botnets to operate IoT devices and change load demands.

With graduate deployment of edge computing in IoT clouds [1], we attempt to provide an initial understanding of feasibility, benefits, and overhead of using edge computing as an infrastructure foundation for future security designs against IoT-based attacks. Specifically, through illustrative examples, we attempt to answer the following questions:

Will edge computing infrastructure be able to host power grids' applications? Hosting power grids' applications appropriately ensures that we can expect few false-positive alerts when no attacks occur, serving as a prerequisite to host security designs. Unlike mobile devices found in general-purpose clouds, the physical location of IoT devices used in power grids is comparatively static; dynamic variations of physical data, which experience significant uncertainty, are critical for grids' operations. To answer this question, we discuss how

edge computing can host energy management systems of traditional power grids. Many research work proposed decentralized algorithms; a few discussed an appropriate computing infrastructure that can run the algorithms efficiently. Hosting energy management in edge computing infrastructure allows monitoring physical states in close proximity to IoT devices, providing a foundation for other security designs.

What security benefits can edge computing bring to power grids? Edge computing indirectly changes information flows from IoT devices, providing an opportunity to design a multi-layer defense mechanism against IoT-based attacks. Using decentralized edge servers to host security designs can meet trade-off of existing perimeter-based protections in IoT clouds that lack the consideration of physical states and centralized intrusion detection systems (IDSs) that suffer from a long latency. In edge servers, we can dynamically deploy a fine-grained policy for each IoT device according to its role in power grids' control operations.

What are the potential overheads with edge computing? Edge computing changes network flows. To fully understand the impacts of the proposed data-centric edge computing infrastructure on IoT clouds and power systems' IT/OT networks, we developed a cyber-physical testbed integrating six different power grids. We implemented a fine-grained access control in this testbed; our preliminary results show that the performance overhead is negligible, with less than 5% on average.

II. Research Gaps in Related Work

Physical Protection. Huang et al. demonstrate that existing physical protection in substations, e.g., primary and secondary control, can remedy impacts of IoT-based attacks, even though they may not completely remove the cyber threat [5]. Soltan et al. further adjust the physical protection by allocating more operational margins, e.g., reserved generation, to respond to sudden and significant changes in load demands caused by IoT-based attacks [6]. These passive approaches do not remove unauthorized access to IoT devices. When adversaries change attack strategies in IoT clouds, these approaches need to adjust physical protection correspondingly, introducing significant overhead on existing control operations.

Policy Enforcement on IoT Devices. Current security approaches detect and prevent attacks on IoT devices by specifying security policy related to their behaviors [7]. The security policy can effectively specify access control by restricting the inbound and outbound traffic of IoT devices [8]. However, those approaches do not enforce safety behavior under IoT-based attacks, as they usually lack the knowledge of physical states of power grids. For example, an access control policy can restrict the number of devices that adversaries can access, but IoT-based attacks can introduce a "sudden" increase of load demands by accessing a small number of high wattage load units, to cause physical disruptions.

Edge Computing in Smart Grids. Some work proposes to apply edge computing into IT networks of power grids [9], to manage the increasing number of physical measurements hierarchically and with a decreased latency. Edge computing is a feasible solution especially for power distribution networks, where utilities deploy off-the-shelf IT/OT components [10]. Similar to [11], we propose to deploy a data-centric edge computing in IoT clouds, not power grids' IT networks. However, unlike [11] providing energy services to individual users, data-centric edge computing collects physical measurements from multiple sites to monitor power grids' security conditions.

III. Building Data-Centric Edge Computing to Enforce Security Policy for IoT-based Attacks

We present in Figure 2 design components of a data-centric edge computing infrastructure to enforce security policies. We first present assumptions of system and threat model. In Section III-A, we present a design of data-centric edge computing to host power grids' applications, avoiding false alerts during attack-free operations. We further discuss security policies to restrict behaviors of IoT devices to ensure grids' safety conditions in Section III-B.

System & Threat Model. We assume that IoT clouds use edge servers at network perimeters to handle services from IoT devices, i.e., high wattage load units, which are shown in Figure 1. This assumption is compatible with existing perimeter-based defenses proposed for IoT clouds. Under this assumption, a communication link connecting a cloud server and IoT devices is divided into three pieces. With a short geographical distance and sufficient bandwidth, Edge-to-Client links enable a short latency and reduce the workload of Cloud-to-Edge links. Some applications also require Edge-to-Edge communication to exchange information from different regions without involving a cloud server.

We assume that adversaries can compromise mobile applications and computers with web interfaces to control IoT devices, e.g., spoofing malicious measurements from IoT devices and/or changing control commands issued by users. We do not assume any forms of software bugs or protocol vulnerabilities granting adversaries such capability. Unlike previous designs of IDSs relying on the integrity of load units, we assume that adversaries have already compromised IoT devices and installed bot applications there. Consequently, adversaries can also use a control and command server to operate on a large number of IoT devices.

A. Data-Centric Edge Computing to Host Power Grids' Applications

It is critical for the data-centric edge computing to host power grids' applications, monitoring physical states collected by corresponding IoT devices. This capability avoids false alerts in normal operational conditions and is a prerequisite to implement security policies for IoT-based attacks.

Control applications used by power grids can be very different from ones used in general-purpose IoT clouds. For example, in an IoT cloud where mobility is a common feature of end devices, a cloud server and edge servers aim at optimizing resource allocation of end devices when they change their locations. In power grids, IoT devices rarely relocate. However, the variations of physical data collected by them play a critical role in control applications. Consequently,

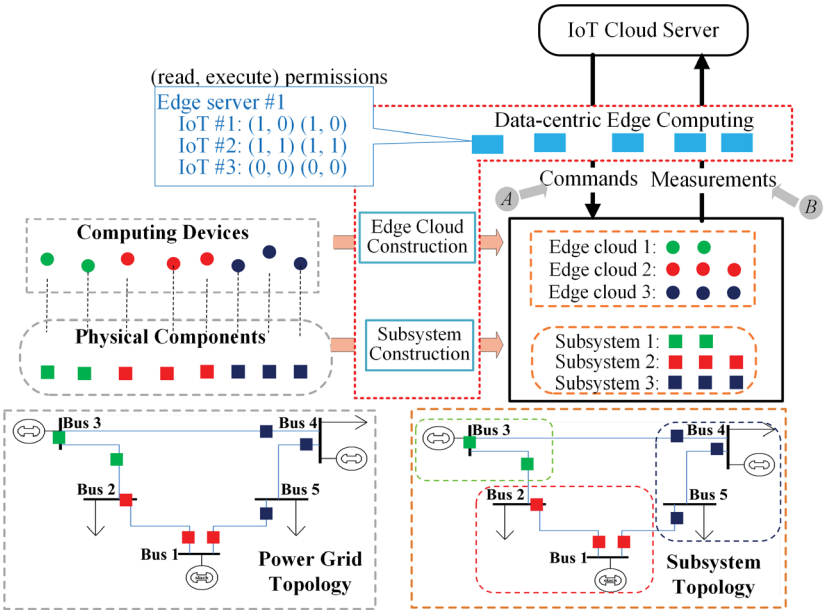


Fig. 2. **Infrastructure overview of** data-centric edge computing for power grids.

when applying edge computing in power grids' computing environment, we should shift the design focus to *classifying data according to its runtime state and service requirements of control operations that are critical to grids' safety condition*.

To reflect the changes of design focus on power grids, we use Figure 2 to present an overview of the data-centric edge computing infrastructure, which includes two procedures:

- ***Subsystem construction.*** In this procedure, we logically group IoT devices or load units in substations (or "buses" used in Figure 2) into different subsystems of a power grid. Consequently, we can use limited computational resources in each edge server to analyze physical states of a small-scale subsystem. The detailed procedure of subsystem constructions varies with control applications as well as the current state of physical components. For example, if we build subsystems for distributed state estimation (shown in the following paragraph), we construct subsystems according to the decomposition step in the algorithm: up to m load units forming a connected sub-graph of the transmission or distribution network of the original power grid are included in a subsystem (where m is a design parameter).
- ***Edge cloud construction.*** To construct an edge cloud (i.e., a small cloud environment including an edge server and IoT devices), we connect IoT devices to an edge server according to the correlations of their collected physical states, determined by the corresponding subsystem. This procedure is different from edge cloud constructions used for general-purpose applications, where end devices are usually connected to an edge server according to their geographical locations. The number of edge servers depends on the available computing resources in IoT clouds and the amount of data that each edge server can process, which is related to the size of a subsystem.

The data-centric edge computing is not necessarily dependent on software-defined networking (SDN), an advanced network technology allowing runtime manipulation of network flows. However, global network visibility and flexible programmability can significantly benefit edge computing [12]. For example, we can use an SDN control plane to extract application-layer payloads related to physical states to classify IoT devices, without proprietary instrumentation on them.

Hosting Power Grids' Applications. Energy management systems use different applications. In this paper, we use the proposed data-centric edge computing to host state estimation, which is a foundation of many of those applications, e.g., optimal power flow analysis and contingency analysis.

Specifically, we apply distributed or hierarchical state estimation methods and use each edge server to host a state estimation for a small region of a power grid. Distributed state estimation usually follows a decomposition-coordination scheme [13]. In the *decomposition* part, a power grid is split into multiple regions, and each region performs its own state estimation independently. In the *coordination* part, measurements of tie lines—transmission lines connecting two different regions—are updated iteratively based on states estimated at each involved region.

The data-centric edge computing can serve as an infrastructure for distributed state estimation. Specifically, to support the decomposition part of distributed state estimation, we can construct a subsystem dynamically by including any number of physical components that form a connected sub-graph of the original transmission or distribution networks, providing more flexibility than previous approaches [13]. To support the *coordination* part, Edge-to-Edge links in edge computing can ensure data exchange, such as tie-line measurements, among different subsystems. By applying the data-centric edge computing, we expect to significantly reduce traffic volumes that need to be sent to a control center and the latency of state estimation, allowing for instant responses on anomalies

in remote sites.

The data-centric edge computing can also host future microgrid applications controlling distributed energy resources (DERs) without involving a control center [14]. Many energy management systems propose decentralized algorithms, e.g., alternating direction method of multipliers (ADMM), to fully explore the benefits of distributed computing [15]. By running those algorithms in different edge servers, we can monitor physical states of future microgrids and specify security policies for those environments correspondingly.

B. Security Policy to Defend IoT-based Attacks

For a better explanation, we classify attacks that can be launched in IoT clouds into two types based on the most likely targets (marked as *A* and *B* in Figure 2) [16]. We discuss how the data-centric edge computing can benefit defenses against these two types of attacks respectively.

- ***Type A: control-related attacks (CRAs).*** Adversaries maliciously change physical state by issuing or modifying control commands to a large number of IoT devices, e.g., dramatically adjusting load demands.
- ***Type B: false or bad data-injection attacks (FDIAs).*** This type of attacks is a severe threat for state estimation used in power systems as adversaries can use knowledge of a power grid, such as topology of transmission or distribution networks, to intelligently compromise physical state without being noticed [17].

The data-centric edge computing can serve as a security middle-box: *each edge server can enforce fine-grained security policy* on connected IoT devices, e.g., "read" and "execute" permissions on measurements and control commands. In Figure 2, we provide a motivation example. We specify permissions for three IoT devices connected to edge server 1; for each device, we put permission bits into two groups, according to the subject issuing operations, e.g., the edge server and the cloud server. In the example, we allow both edge server and cloud server to read measurements from device 1 but not to execute commands on it.

Protection Against Control-related Attacks (Type A). Even though there are many IDS approaches defending the control-related attacks, most of them are proposed for a centralized unit (such as a control center) and rely on the rich information collected there. Centralized IDSs introduce three problems. *First*, they need to put trust on the wide variety of computing devices in power systems, complicating threat model and making deployment of the IDSs less practical. *Second*, they can suffer from time-of-check-to-time-of-use (TOCTTOU) vulnerability as there is a wide-area network in which adversaries can compromise control commands after analysis in centralized IDSs. *Last*, when a successful detection requires a response to remedy physical damage, there is another round trip of communication for response mechanisms to reach remote substations.

Setting "execute" permissions can significantly restrict the range of IoT devices, to which an adversary can issue malicious control commands. Using the example in Figure 2, even if an adversary can compromise a critical device to inject commands, he/she will not be able to maliciously reroute the command to device 1 or 3 as the edge server does not allow commands being executed in them. Consequently, adversaries can only inflict disruptions on device 2 even if they successfully inject malicious commands. Furthermore, if we can ensure that the target power grid can tolerate any commands executed in device 2, we can completely prevent physical damage from adversaries.

More importantly, we envision that the data-centric edge computing can remedy the drawbacks of centralized IDSs. *First*, because each edge server, distributed in different sites, does not require complicated implementation, we can trust edge servers instead of power systems, reducing the complexity of threat model and preventing a control center from becoming a single point of attacks. *Second*, with edge servers in close proximity to IoT devices, we can reduce the width of the TOCTTOU window as well as latency to launch response mechanisms.

Protection Against False-data Injection Attacks (Type B). In the data-centric cloud computing, we use edge servers to host state estimation application to monitor physical states of subsystems. At first glance, it seemed that edge servers can be vulnerable to FDIAs. However, "read" permission set by edge servers can help make the attack challenging to succeed.

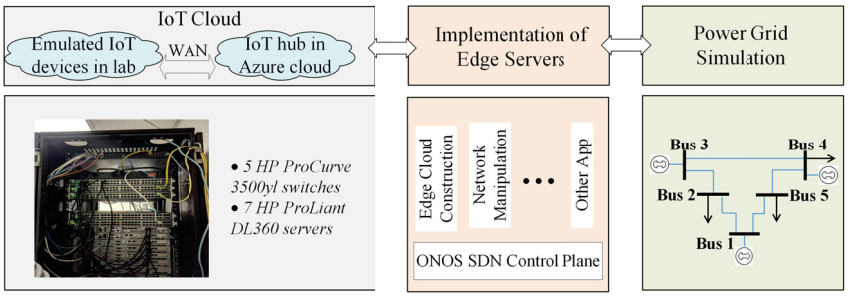
Previous works show that randomly filtering compromised data (even only a small part) can be effective against false data injection attacks, as remaining compromised data can trigger alerts in state estimation [18]. However, there lacks a friendly way to implement those mechanisms: they either require modifying state estimation software, which can reduce estimation accuracy, or intentionally introducing physical disruptions.

The "read" permission can directly enable such functionality without any software modification and physical disruptions. Specifically, by randomly marking some data from IoT devices as "unreadable," an edge server uses a set of data, which is challenging for adversaries to obtain, to perform state estimation. Without such knowledge, randomly compromising data can easily trigger alerts. In addition, we can set "read" permissions based on data redundancy, to maintain the accuracy of state estimation. Because an edge server can connect multiple devices, achieving wider observability than a single device, it can identify redundancy of data and thus select a subset of data sufficient enough for accurate state estimation (also known as a basic or critical data set).

Another byproduct of setting "read" permissions is to protect data privacy and user confidentiality. By applying the edge computing in power grids, we can further determine a user's confidentiality based on physical applications in which the user is involved.

IV. Preliminary Evaluation on Cyber-Physical Testbed with Edge Computing Infrastructure

To provide an initial feasibility evaluation of the data-centric edge computing, we develop a cyber-physical testbed shown in Figure 3. This testbed includes realistic cyber (edge computing infrastructure based on SDN) and physical (power system analysis) aspects of six different power grids, being used as evaluation cases.



Cases Implemented in the Cyber-Physical Testbed

Case	*Power Grid Simulation*	*Communication Network*
Case 1	IEEE 24-bus	DataX (12)
Case 2	IEEE 30-bus	Abilene (22)
Case 3	RTS96 73-bus	HurricaneElec (48)
Case 4	Poland 256-bus	Chinanet (84)
Case 5	Poland 406-bus	Cesnet (156)
Case 6	Poland 1153-bus	Forthnet (248)

Fig. 3. **Cyber-physical testbed with edge computing infrastructure.**

- ***IoT Clouds.*** We implemented an IoT cloud by connecting an IoT hub application in Microsoft Azure cloud and emulators of IoT devices in our lab network, which is built on five HP ProCurve 3500yl switches and 7 HP ProLiant DL3600 servers. Each switch has 48 ports, and we extended each server with 20 Ethernet ports. By grouping switch ports into different VLANs (virtual local area networks), we built six networks of different sizes (we include the number of switches of each network in parentheses) based on topology in TopologyZoo dataset [19].
- ***Power Grid Simulations.*** To provide physical data for network traffic, we simulate power systems with different transmission or distribution networks in MATPOWER, an open-source MATLAB toolbox [20]. MATPOWER, analyzing steady state of power grids, plays two critical roles. *First*, it generates and delivers physical data to emulated IoT devices via a separate communication channel; IoT devices further issue the data to the IoT hub application via the IoT cloud. *Second*, when there is an IoT-based attack, MATPOWER estimates physical consequences of load-changing commands delivered by the attack and updates the physical data with the IoT devices.
- ***Edge Servers.*** We followed suggestions in [12] and implemented edge computing infrastructure on top of SDN. Specifically, we used ONOS SDN controller to manipulate network traffic and group IoT devices into different edge servers based on traffic contents. ONOS provides two features that are critical to implementing functionality proposed in this paper: (i) it allows developing applications, e.g., security policies, and adding them to the "core" engine at runtime; and (ii) it facilitates adding parsers and encoders of new network protocols so that controllers are able to obtain knowledge related to power grids from runtime network packets.

In preliminary evaluation shown in Figure 4, we focused on the overhead of edge servers constructing subsystems and grouping IoT devices and performance of security policies presented in Section III-B. In future work, we will add implementation and evaluation of other security policies on top of the proposed edge computing infrastructure.

Using left y-axis and bar graphs in Figure 4, we present the overhead of determining subsystems and grouping IoT devices for edge servers. For each case (specified by the x-axis), we increase the maximum size of subsystems from 4 to 10 substations and measure the impact on overhead. We can see that the latency to construct subsystems is around 1.6 milliseconds (ms) on average, which is neither significantly affected by the size of subsystems nor the size of simulated power grids.

Although the overhead of configuring edge servers is not affected by the size of communication networks, it increases with the size of subsystems, because an SDN controller monitors and manipulates network traffic from multiple IoT devices. In our experiment, we used a single instance of ONOS SDN controller to connect all switches in the simulated network. When we increased the scale of subsystems to include up to 10 substations, congested networks increased the latency for the SDN controller to configure edge servers. Even in this case, we could still configure edge servers under 6 ms on average. In practice, we can deploy multiple controllers to further reduce overhead.

Using the right y-axis and line plots in Figure 4, we present the goodput of the SDN controller to measure its capability to execute the proposed security policy. Specifically, we used the SDN controller to monitor the load-changing commands exchanged between the IoT hub application and the emulated IoT devices and restrict the number of network packets such that delivered commands would not introduce physical damage in power grids. We compare the goodput when the SDN controller is equipped with and without the security policy (with 95% confidence interval).

In our experiment, goodput varies from 4 to 7.5 Mbps. By measuring elapsed times on the block of codes that implement the security policy, we expect that enabling the security policy would increase around 5 to 10 ms on the round trip time between the SDN controller and switches, further reducing goodput. However, such impacts due to the security policy is less significant compared to the impacts caused by congested networks, as we have made heavy usage of available network bandwidth in our lab network and goodput experiences significant variations from time to time and in different evaluation cases. By analyzing network traces in detail, a certain number of re-transmission of network packets largely contributed to such variation. Even under such busy network communication, we can still obtain at least 4 Mbps of goodput.

V. Conclusion

In this paper, we present the design of a data-centric edge computing infrastructure used for IoT devices in power grids. By equipping edge servers with knowledge of power

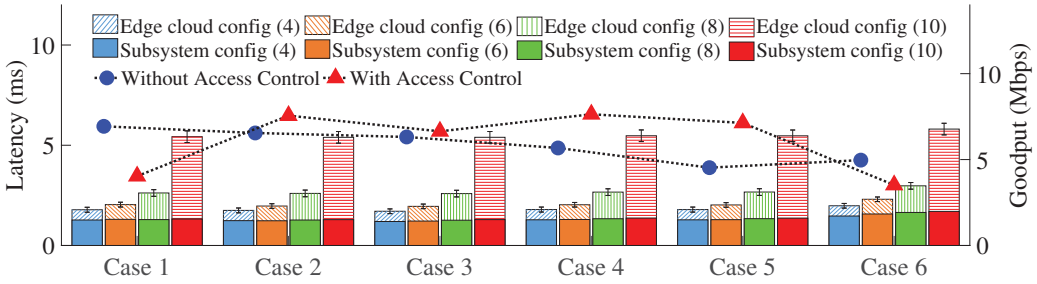


Fig. 4. **Overhead of the data-centric edge computing.** The left *y*-axis with bars shows the overhead to construct subsystems and group IoT devices while the right *y*-axis with the line plots shows the goodput of SDN controller executing security policies (with 99% confidence interval).

grids in different subsystems, we can enhance the proposed edge computing with security policies to defend IoT-based attacks, e.g., by filtering compromised measurements and/or restricting the range of adversaries' activities. Preliminary evaluations show promising results that edge servers can efficiently process physical data and manipulate network traffic at runtime. In future work, we will design and implement security policies on the data-centric edge computing and evaluate their performance over different IoT-based attacks.

References

[1] S. Muyeen, and R. Saifur. "Communication, control and security challenges for the smart grid," Institution of Engineering and Technology, 2017.

[2] S. Soltan, P. Mittal, and H. V. Poor. "BlackIoT: IoT botnet of high wattage devices can disrupt the power grid," *Usenix Security*, 2018, pp. 15-32.

[3] K. K. Yap, M. Motiwala, J. Rahe, S. Padgett et al. "Taking the edge off with espresso: scale, reliability and programmability for global Internet peering," *Conference of the ACM Special Interest Group on Data Communication*, 2017, pp. 432-445.

[4] D. Puthal, M.S. Obaidat, P. Nanda, M. Prasad et al. "Secure and Sustainable Load Balancing of Edge Data Centers in Fog Computing," *IEEE Communications Magazine*, vol.56, no.5, pp.60-65, 2018.

[5] B. Huang, A. A. Cardenas, and R. Baldick. "Not everything is dark and gloomy: power grid protections against IoT demand attacks," *Usenix Security*, 2019, pp. 1115-1132.

[6] S. Soltan, P. Mittal, and H. V. Poor. "Protecting the grid against mad attacks," *IEEE Transactions on Network Science and Engineering*, 2019.

[7] Z. B. Celik, G. Tan, and P. D. McDaniel. "IoTGuard dynamic enforcement of security and safety policy in commodity IoT," *Network and Distributed System Security Symposium (NDSS)*, 2019.

[8] W. He, M. Golla, R. Padhi, J. Ofek et al. "Rethinking access control and authentication for the home Internet of things (IoT)," *Usenix Security*, 2018, pp. 255-272.

[9] F. Okay and S. Ozdemir. "A fog computing based smart grid model," *International Symposium on Networks, Computers and Communications (ISNCC)*, pp.1-6, 2016.

[10] Y. Yan and W. Su. "A fog computing solution for advanced metering infrastructure," *IEEE/PES Transmission and Distribution Conference and Exposition (T&D)*, pp.1-4, 2016.

[11] M. Faruque and K. Vatanparvar. "Energy Management-as-a-Service Over Fog Computing Platform," *IEEE Internet of Things Journal*, vol.3, no.2, pp.161-169, 2016.

[12] Y. C. Hu, M. Patel, D. Sabella, N. Sprecher et al. "Mobile edge computing—A key technology towards 5G," *ETSI white paper*, 2015, pp. 1-16.

[13] J. A. Aguado, C. Perez-Molina, and V. H. Quintana. "Decentralised power system state estimation: A decomposition-coordination approach," *Porto Power Tech*, 2001, pp. 6-pp.

[14] R. Lasseter, A. Akhil, C. Marnay, J. Stephens et al. "Integration of distributed energy resources: the CERTS Microgrid Concept," *Consortium Electric Reliability Technology Solution*, 2002.

[15] W. J. Ma, J. Wang, V. Gupta, and C. Chen. "Distributed Energy Management for Networked Microgrids Using Online ADMM With Regret," *IEEE Transactions on Smart Grid*, 2016, pp. 847-856.

[16] H. Lin, H. Alemzadeh, D. Chen, Z. Kalbarczyk et al. "Safety-critical cyber-physical attacks: analysis, detection, and mitigation," *the Symposium and Bootcamp on the Science of Security*, 2016, pp. 82-89.

[17] Y. Liu, P. Ning, and M. K. Reiter. "False data injection attacks against state estimation in electric power grids," *ACM Transactions on Information and System Security*, 2011, p. 13.

[18] M. A. Rahman, E. Al-Shaer, and R. B. Bobba. "Moving target defense for hardening the security of the power system state estimation," *ACM Workshop on Moving Target Defense*, 2014, pp. 59-68.

[19] S. Knight, H. X. Nguyen, N. Falkner, R. Bowden et al. "The Internet topology zoo," *IEEE Journal on Selected Areas in Communications*, 2011, pp. 1765-1775.

[20] R. D. Zimmerman, C. E. Murillo-Sánchez, and R. J. Thomas. "MATPOWER: Steady-State Operations, Planning, and Analysis Tools for Power Systems Research and Education," *IEEE Transactions on Power Systems*, 2011, pp. 12-19.

Bibek Shrestha is a Ph.D. student in Computer Science and Engineering department at the University of Nevada at Reno. He obtained his bachelor's degree in Computer Engineering from Tribhuvan University, Nepal. His research interest includes cybersecurity, Internet of Things (IoT), high performance computing etc. He can be reached at bibek.shrestha@nevada.unr.edu.

Hui Lin is an Assistant Professor at the Computer Science and Engineering Department in the University of Nevada at Reno. He earned his Ph.D. degree from the University of Illinois at Urbana-Champaign in 2017 in electrical and computer engineering. His research interests include cyber security, intrusion detection systems, and software-defined networking (SDN) in the areas of cyber-physical systems, such as power systems. He has successfully adapted Bro, a runtime network traffic analyzer, to support network protocols (e.g., DNP3) commonly used in power grid infrastructure. The DNP3 analyzer that he developed has been included in Bro and can be downloaded freely by utility companies. His current work focuses on applying SDN in cyber-physical systems; he intends to use SDN's network programmability to design flexible cyber-physical systems which can quickly respond to cyber-attacks and accidents. He is a member of IEEE. Contact him at hlin2@unr.edu.