Towards Secure Demand-Response Systems on The Cloud

Apurva Mohan
Honeywell Research Labs
Golden Valley, MN
Email: Apurva.Mohan@Honeywell.com

Daisuke Mashima
Fujitsu Laboratories of America, Inc.
Sunnyvale, CA
Email: dmashima@us.fujitsu.com

***Abstract*—Demand response (DR) systems are gaining fast adoption and utilities are increasingly relying on them for peak load shaving, demand side management, and maintaining power quality. DR systems are cyber-physical systems (CPS) where the communication component is cyber, whereas the control components have physical effects. As DR systems experience wider adoption and manipulate much larger loads, achieving scalability has become an important concern. On the other hand, demand response events are often sporadic, and maintaining systems and infrastructure that could easily scale up or down is often desirable for utility companies in terms of operational cost, which makes us envision that DR systems would eventually move to the cloud. However, moving to cloud is not an elixir as it brings some concerns of its own. In this paper, we focus on OpenADR 2.0-based systems and discuss security properties and challenges that must be considered when migrating DR systems to the cloud.**

***Keywords*-Cloud; Demand Response; Information Security; Security Threats; Access Control; Communication System Security; Data Security;**

I. Introduction

Demand response (DR) and demand side management (DSM) are quickly becoming mature technologies on which utilities are increasingly depending for load balancing and sustainable grid operations. These technologies create a win-win situation for the utilities and electricity consumers by providing cost savings to the utilities and cash discounts to the consumers for either rescheduling or reducing their loads [1]. While in the past DR was done by means of manual human intervention, e.g., by requesting demand reduction using phone call, fax, or email, the two-way communication capability of the smart grid between the utility and the consumers has enabled automated demand response and demand side management technologies. An internationally recognized standard for automated demand response, which is recently attracting significant attention, is OpenADR 2.0 profile specification [2]. In this paper, both DR and DSM are jointly referred to as Demand Response (DR) unless differentiated. In DR, the utility determines when it would be useful for the consumers to either lower or shift their electric demand to improve load balancing on the grid. Consumers register for DR programs that are run by utilities. They use a client device referred to as a DR client to communicate with the utility DR entity called the DR server. When DR programs are activated, the DR server sends initiation commands to its DR clients. The DR clients are in turn responsible for dropping the loads connected to them for the agreed upon period. If the loads are programmable, they can be shifted in time, too.

DR systems are becoming part of the critical infrastructure for electric power since, for instance, they can be often seen as virtual generators. DR program management and communications use cyber communication technologies like the Internet to enable command-response exchange between the DR server and client. The DR clients interact with the physical electric equipment to shed load or to programmatically alter their loads. In industrial sites, the equipments that the DR clients manipulate are often part of safety-critical processes, so controlling the DR clients via the Internet makes the operations very sensitive. OpenADR 2.0 profile specification [2] addresses communication security for automated DR effectively.

DR is growing very rapidly and is seeing wide adoption in many countries, including the U.S. For instance, it is expected that the number of sites that are capable of automated demand response will grow from 200,000 to 2 million in the next 10 years [3]. DR growth projections in the U.S., Asia Pacific, and Europe are very promising and it is predicted that DR programs could potentially manipulate up to 88GW or 20% of the peak electric load in the U.S. by 2019 [4]. Although this growth is very encouraging, it comes with a major challenge - scalability. On the other hand, in many demand response programs run by U.S. utility companies, demand response events are called only a couple of times a year, so maintaining the infrastructure with such scalability all the time is not desired in terms of the operational cost.

To address the demands in scalability and elasticity, DR program operations are now being moved to the cloud. Although this migration contributes to mitigation of scalability challenges and reduction of operational and installation cost on utility companies, it introduces several new ones. Among them, we focus on security and will present challenges for DR operations in the cloud. While [5] discusses the security issues when deploying power grid system applications in general, to the best of our knowledge, ours is the first attempt that elaborates security requirements and challenges specifically in cloud-based demand response systems. Also,

the OpenADR [2] addresses communication security for DR, which is applicable to the cloud scenario as well, but it does not cover security issues that would arise in the cloud hosting end. A framework to introduce security considerations in smart grid standards in a comprehemsive manner can be found in [6]. It proposes mapping security objectives, requirements, mechanisms, and residual risks for technologies covered in the smart grid standard. This framework can be used to integrate the newly proposed security considerations into existing smart grid standards.

The rest of the paper is organized as follows - Section II presents security requirements and challenges for automated demand response systems which are desired in any cloud-based deployment. Section III presents overview of OpenADR 2.0 profile specification, which has the broadest adoption among automated DR standards [7]. After that, Section IV discusses implementation options on typical cloud service models, followed by discussion for secure cloud-based DR systems in Section V. Finally, Section VI concludes the paper.

II. Security Requirements and Challenges for DR Systems

A. Requirements Derived from Regulation

In this section we will present cybersecurity requirements for DR derived from a regulatory authority. These requirements have been extracted out of North American Electric Reliability Corporation - Critical Infrastructure Protection (NERC-CIP) standards - CIP-002-1 through CIP-009-2 [8]. Although the NERC-CIP requirements do not immediately apply to the DR systems at present because they do not manipulate enough load, in future DR programs would manipulate much higher making them subject to NERC-CIP compliance. Also, we base our discussion on NERC-CIP because it is one of the primary standards for electric reliability in the U.S.

Any organization that is trying to achieve compliance with NERC-CIP standards would apply them to the assets that are within their organization perimeter. In DR systems, critical assets would normally be spread across three organizations: utilities that initiate DR programs, DR providers and aggregators, and DR clients that actually perform load control. The critical DR assets with each of these organizations may include - i) Utilities - Control stations, communication channels, load forecasting and time-of-use pricing systems; ii) DR providers - DR server, communication channels; and iii) DR customers - DR clients, communications channels, electrical loads.

The cybersecurity requirements in NERC-CIP that we should consider for DR systems are presented below -

1) ***Electronic Security Perimeter:*** Every critical cyber asset should be within the electronic security perimeter.
2) ***Identity Management:*** The DR system should provide digital identity management to all internal and external entities. This is important for all communication, coordination, and control activities within a single DR server or when a number of these servers collaborate.
3) ***Access Control:*** Appropriate access control should be enforced to mediate access to - i) All critical assets; and ii) Access to all electronic access points on the perimeter. This includes implementation of a secure authentication mechanism.
4) ***Information Protection:*** Appropriate measures should be taken to identify, classify, and protect sensitive information associated with DR operations and communications.
5) ***Critical Asset Protection:*** Critical assets should be identified and protected from damage from actions of remote entities. These mechanisms should work in conjunction with the safety features of identified critical assets.

B. Required Security Properties against Cyber Threats

In this section, we will overview the general information security guideline for automated demand response systems. Our discussion below is derived out of well-known CIA (Confidentiality, Integrity, and Availability) Triad as well as cybersecurity framework for critical infrastructure published by NIST [9].

1) ***Confidentiality:*** In demand response systems, privacy-sensitive customer data has to be stored on servers. Data may include: fine-grained, real-time electricity usage data collected by smart meters, billing and personal information, demographic data etc. to perform accurate demand peak prediction, negawatt prediction, and optimal portfolio selection. Such data must be encrypted on the network during transit and also in the storage while at rest to prevent unauthorized accesses to the contents.
2) ***Integrity:*** Automated demand response requires communication over network to convey demand response event information, including how much electricity has to be curtailed and when. If such information would be maliciously modified or forged, it may pose significant impact on the grid and could potentially cause instability and potentially even blackouts. Thus, integrity of data communicated is important. In addition, data used in the DR system, for example customers' usage data for peak demand prediction, also has to be integrity protected to avoid malicious manipulation.
3) ***Availability:*** Demand response, especially fast-DR, sometimes requires timely communication to control the electricity demand within a very short response time. Thus, real-time availability of communication channels and server systems is crucial.

4) ***Authentication:*** Related to integrity, it has to be guaranteed that only an appropriate party (e.g., a utility company or a DR aggregator that a DR program participant has established a service contract with) can issue DR event signals. This can be achieved by robust sender authentication. On the other hand, DR clients (customers) have to be authenticated to ensure that only participating customers receive DR signals and also that curtailment reports are actually sent by DR clients whose identities are claimed in the report payloads.
5) ***Non-repudiation:*** Demand response usually establishes service contract between a service provider (a utility or DR aggregator) and a customer, which defines for instance the rule for calculating monetary incentives to be paid to customers. To enable each participating entity to later dispute or challenge against other entities, verifiable evidences of all interactions should be kept.
6) ***Auditing and Logging:*** To counter cyber attacks from external attackers and attacks mounted by insiders, reliable auditing is imperative. Auditing is facilitated by secure logging of events and sensitive operations. Additionally, log data has to be secured against unauthorized access and tampering.

III. Brief overview of OpenADR 2.0

OpenADR 2.0 profile specifications [2] are being developed by the OpenADR Alliance to define a standardized communication model for automated demand response, including the messaging scheme used between a demand response (DR) automation server, for example at a utility, and DR participants. The latest version of the specification is OpenADR 2.0B profile specification that was released in mid-2013. OpenADR is expected to be a dominating mechanism at least for the next 10 years, contributing to lower product development costs [3].

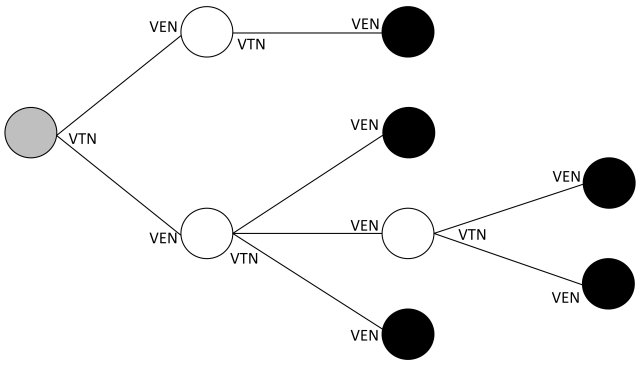

Figure 1. Example of tiered architecture of OpenADR nodes.

OpenADR is designed based on the subset of OASIS Energy Interoperation (EI) Version 1.0 [10] for automated demand response. The OpenADR 2.0A profile specification defines the feature set for simple devices (e.g., thermostats), while the OpenADR 2.0B profile is designed for full-feature energy management solutions including DR aggregation. OpenADR defines a communication model between a DR server, also referred to as a virtual top node (VTN), that would typically reside in a utility or other types of demand response service providers, and DR clients, also referred to as virtual end nodes (VENs). Nodes are organized in a tree-like structure and are categorized into VTNs or VENs. Some entities mediating DR communications and transactions, such as DR aggregators, may implement both VTN and VEN functionalities. An example of such an architecture is shown in Figure 1.

The security scheme implemented in OpenADR largely relies on well-established standards. OpenADR mandates all nodes (both VTNs and VENs) to have unique public key certificates, and TLS (Transport Layer Security) version 1.2 [11] with client authentication is used to ensure mutual authentication as well as message integrity and confidentiality. In addition, XML Signature [12] can be optionally used for non-repudiation. These security mechanisms will be used as the baseline in the cloud-based DR, which is the focus of this paper.

IV. Cloud-based DR System Architecture

We next present the cloud operations model for DR. A high-level conceptual architecture is shown in Figure 2. The Utility Operations Center (UOC) is the main command and control entity in the utility. The DR server is located in the cloud and uses web technologies for communications. The DR servers may be connected directly to the customers or to intermediate aggregators as shown in Figure 2. These aggregators may be in the cloud or outside of it. The DR aggregators connect to various types of DR clients. Figure 2 illustrates three types of DR clients; first consisting of industrial DR customers that typically use OpenADR 2.0B profile and second, consisting of home or light commercial DR customers that often use OpenADR 2.0A profile, and third that are deployed in the cloud itself and communicate with the industrial facility using smart energy protocols like Smart Energy Profile 2.0 [13] or ECHONET Lite [14].

In the cloud, the DR server can be deployed in various ways. We illustrate one such option using Microsoft Azure service bus [15]. Figure 2 shows the high level Azure service bus architecture in the dashed box, which shows the server deployment internals. The server contains three main components, namely Control, Analytics, and Event Store, that will be mapped to OpenADR service components. DR Topic is the communication interface that receives communications from DR clients (often mediated by the aggregators) and the sub boxes represent the interfaces that communicate out to the DR clients. This architecture provides the DR server with a very high fan-in and fan-out capability enabling it

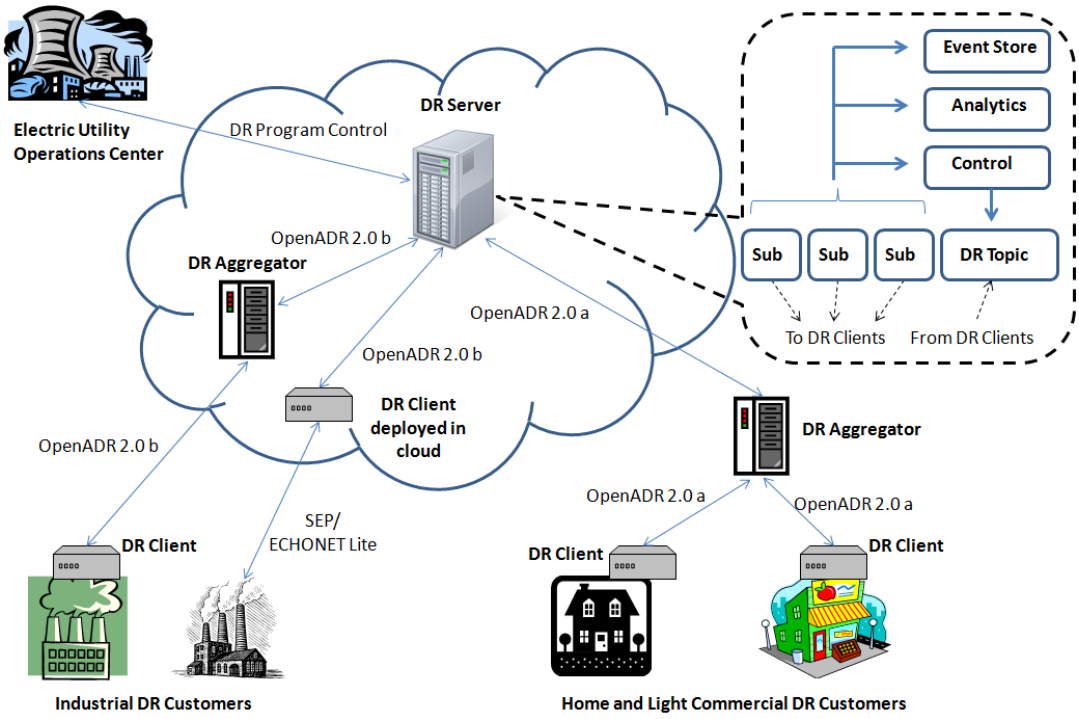

Figure 2. Conceptual architecture of cloud based DR operations.

to interface with a very high number of DR clients for each instance of the DR server. It is envisioned that for scalability and fault tolerance, the DR servers will be replicated. More DR servers can be added almost instantaneously in the cloud since the DR server functionality will logically be encapsulated in software.

In residential DR market, some companies, including startups, have already launched SaaS (software-as-a-service) DR services [3]. These companies employ standards-based automated DR solutions leveraging third party cloud platforms. Additionally, some cloud-based deployments take advantages of big data analytics for energy that can be leveraged for improving the performance of DR programs. Although these companies may be using different models than the Azure service bus model described in this section, the basic concepts of cloud deployment of DR servers remain the same.

V. Security Discussion

In this section we will discuss the security of cloud-based DR systems, focusing on systems using OpenADR overviewed in Section III.

A. Security for DR Communication

The security specification defined in OpenADR 2.0 profile specifications requires TLS with client authentication for DR communications. This mechanism is an effective solution for communications security even when the deployments migrate to cloud. Use of XML Signatures are also proposed for non-repudiation as an optional measure, which is an effective solution.

One potential risk to be considered would be the management of a private key of each node. Namely, a cloud service provider on which VTNs and/or VENs are deployed could misuse it. However, as long as the key is not used for any purpose other than OpenADR communication, the impact caused by cloud providers' misbehavior is limited. In addition, in OpenADR, private keys and corresponding public key certificates are issued by a CA run by OpenADR Alliance (technically a company entrusted by it) and validity of the certificates are limited only within OpenADR communication. Having said that, periodic audit of software module integrity or system users that have access to the keys, periodic regeneration and update of the keys, and continuous anomaly detection in the usage of credentials are desired.

To further raise the security bar in multi-hop OpenADR communication, integration of the mechanism recently proposed in [16] can be considered. The extension allows end-most DR clients to obtain extra information that may be helpful for detecting potentially-malicious intermediaries on the multi-hop DR signal distribution path.

Another type of risk for the cloud-based deployment is that adversaries can target the communication between the utility's system and the DR system on the cloud provider to disrupt operations. Deploying best practices in defending against such attacks, including establishment of authenti-

cated, encrypted communication channel, makes the attackers' work factor much higher.

B. Protecting DR systems

To securely operate DR services, communication security alone is not sufficient. Just like the case where DR system is deployed in utility's enterprise system, system components and modules that implement DR services must be protected against cyber attacks as well as insider threats. In addition, when systems are deployed on the cloud, we need to worry about additional attack surface and adversaries, including other tenants on the same cloud platform and the cloud service provider itself.

In traditional (non-cloud) deployments, cyber vulnerability assessment would require evaluation of the deployment infrastructure of the DR provider, but in the cloud compliance with this requirement becomes tricky. The protection of electronic access points and security perimeters becomes the responsibility of the cloud provider. This means that availability of the communication infrastructure or robustness against DoS attack targeting service availability are largely outside of the utility's control. The DR provider has to rely on the SLA with the cloud provider to maintain compliance. For protecting the critical assets like DR servers, the DR provider should deploy the services in a manner that leverages the security services provided by the cloud provider. The level of control that the DR provider enjoys over its DR assets will depend a lot on whether the DR service is hosted using IaaS or PaaS cloud model. Also, it is hard to audit if the security services/mechanisms that the cloud security provider claims to implement are indeed implemented. For example, what guarantees does the cloud provider give for securing the private keys of DR servers in the cluster and are the guarantees reliable?

The DR servers in a cluster would fan-out to a much larger number, possibly to hundreds of thousands of DR clients. This makes the impact of a single DR cluster compromise much higher as it can now reach many more clients. The challenge is that with the current architecture of the Azure service bus, the sub boxes (refer to Figure 2) are all in the same trust domain, so a security compromise in the domain would seamlessly propagate.

As of now, the DR programs are launched from either the Utility Operations Center (UOC) or a trusted third party DR provider. With the migration to cloud, the cloud provider will also have to be a trusted entity in the DR program execution. This is because, unlike the utility or trusted DR providers, the cloud providers have much larger operations and diverse employee base. It is hard to maintain insider trust and hardened security in such large scale operations. In the absence of high trust, access control for systems or sensitive data becomes difficult, and a rogue employee working for the cloud provider or security breach in the cloud could manipulate a very high load and could potentially impact the local grid stability. Background checks for personnel involved in sensitive activities should be enforced[1], but tightening this trust to the same level as the one in a UOC is very challenging because it required upgrades in technology, process, and people. Implementing reliable, tamper-evident logging mechanism using cryptographic primitives, such as [17], as part of DR system components deployed on the cloud can be considered as one of the solutions.

Migration into cloud changes the attack surface of the system, which requires the re-consideration of requirements for deployment of intrusion detection systems, network monitors, etc. For instance, in case of a private server hosted in-house, the entry point for the attacker is limited to the open network ports. However, on the cloud, physical isolation of the hardware is not always guaranteed. In addition, some functionality or resource could be shared with other tenants.

C. Security and Privacy of Customer Data

To offer demand response services, each electricity customer's energy consumption data, which is usually collected and reported by smart meters outside of a DR system itself, is essential to conduct baseline forecasting, negawatt prediction, and measurement of DR performance for incentive calculation, and so forth. On the other hand, the privacy issues caused by detailed energy consumption data have been reported, e.g., in [18]. Thus, security consideration should also be made on such data utilized for demand response operations. The outcome of carelessly delegating data to a third-party cloud provider could be significant as evidenced by an incident happened in New York utilities [19].

Data has to be protected against other tenants, external entities, and a cloud service provider. While implementing data encryption and access control may address the first two, privacy protection against cloud service providers would require one step further. Ideally, data has to be encrypted even when it is processed on memory, instead of just keeping data at rest encrypted. While recent cryptographic primitives, such as homomorphic encryption [20], accomplish such a goal, it still requires significant overhead for general computation that is often required to apply machine learning techniques. Given the QoS requirements, it is not yet a practical solution.

Security against cloud service providers also include other aspects. They could modify or discard data on the storage, which may affect the quality and availability of DR services. To detect such intentional or unintentional misbehaviors by cloud service providers, a utility's system operator should be able to attest the integrity and availability of data and software. Performing attestation for data and software deployed in the cloud is a very challenging problem [21].

From the perspective of data security, we recommend the DR deployment on a hybrid cloud model. A hybrid cloud,

[1]Typically this is part of the security audit for public cloud providers

as the name suggests, is combination of public and private clouds where some operations stay in the private cloud (e.g., a local data center) of the owner while the others are located in a public cloud service. For instance, while sensitive personally identifiable information (PII), such as name, complete mailing address, and fine-grained energy usage data can be stored locally in utility, minimal, non-sensitive data, including anonymized customer ID, zip code, and lower-resolution energy usage data [22] are sent to the cloud. Demand prediction and other analytics required to run DR service can be done on the cloud, and only aggregated DR performance of each customer can be reported back to the utility along with anonymized ID, which can be used for incentive payment etc.

VI. Conclusion and Future Work

In order to enable massive-scale demand response services, we envision DR operations would move to the cloud in the near future to address scalability issues. Although moving to cloud addresses scalability issues, such deployments would inevitably introduce security issues. In this short paper, we discussed security challenges that we need to consider when migrating DR system into the cloud. Our future research direction includes development of practical implementation guideline for secure cloud-based DR systems and prototype implementation based on it.

References

[1] M. H. Albadi, and E. F. El-Saadany. "Demand Response in Electricity Markets: An Overview", in Proc. of IEEE Power Engineering Society General Meeting. Vol. 2007, 2007.

[2] OpenADR Alliance. Online: http://www.openadr.org

[3] Brett Feldman and Bob Lockhart. "Automated Demand Response", Online: http://www.navigantresearch.com/research/automated-demand-response, 2014.

[4] Federal Energy Regulatory Commission. "A National Assessment of Demand Response Potential", Online: https://www.ferc.gov/legal/staff-reports/06-09-demand-response.pdf, 2009.

[5] Gyorgy Dan, Rakesh B. Bobba, George Gross, and Roy H. Campbell. "Cloud Computing for the Power Grid: From Service Composition to Assured Clouds", in Proc. of the 5th USENIX Workshop on Hot Topics in Cloud Computing (HotCloud 2013), San Jose, CA, 2013.

[6] Apurva Mohan and Himanshu Khurana. "Towards Addressing Common Security Issues in Smart Grid Specifications", in Proc. of the 5th International Symposium on Resilient Control Systems (ISRCS), Salt Lake City, UT, 2012.

[7] Edward Koch and Tariq Samad. "Power Industry Is Embracing Automated Demand Response Standard", IEEE Smart Grid, Online: http://smartgrid.ieee.org/april-2013/841-power-industry-is-embracing-automated-demand-response-standard

[8] North American Electric Reliability Corporation. "North American Electric Reliability Corporation - Critical Infrastructure Protection (NERC-CIP) Standards", Online: http://www.nerc.com/pa/stand/Pages/default.aspx

[9] National Institute of Standards and Technology. "Framework for Improving Critical Infrastructure Cybersecurity", Online: http://www.nist.gov/cyberframework/upload/cybersecurity-framework-021214.pdf, 2014.

[10] "Energy Interoperation version 1.0", Online: http://docs.oasis-open.org/energyinterop/ei/v1.0/energyinterop-v1.0.html, 2012.

[11] Tim Dierks. "The Transport Layer Security (TLS) Protocol Version 1.2," Online: http://tools.ietf.org/html/rfc5246, 2008.

[12] D. Eastlake, J. Reagle, and D. Solo. "(Extensible Markup Language) XML-Signature Syntax and Processing", Online: http://www.ietf.org/rfc/rfc3275.txt, 2002.

[13] "Smart Energy Profile 2.0", Online: http://goo.gl/yrSFgx.

[14] ECHONET Lite Consortium, Online: http://www.echonet.gr.jp/english

[15] Microsoft. "Using Windows Azure Service Bus for ... Things!", Online: http://msdn.microsoft.com/en-us/magazine/jj133819.aspx.

[16] Daisuke Mashima, Ulrich Herberg, and Wei-Peng Chen. "Enhancing Demand Response Signal Verification in Automated Demand Response Systems", in Proc. of the 5th Innovative Smart Grid Technologies Conference, Washington, D.C., 2014.

[17] Scott A. Crosby, and Dan S. Wallach. "Efficient Data Structures For Tamper-Evident Logging", in Proc. of the 18th USENIX Security Symposium, 2009.

[18] Stephen McLaughlin, Patrick McDaniel, and William Aiello. "Protecting Consumer Privacy from Electric Load Monitoring", in Proc. of the 18th ACM Conference on Computer and communications security, 2011.

[19] "New York utilities failed to protect customer information, report finds", Online: http://www.infosecurity-magazine.com/view/26967/new-york-utilities-failed-to-protect-customer-information-report-finds/, 2012.

[20] Craig Gentry. "Computing arbitrary functions of encrypted data", in Communications of the ACM 53.3, 2010.

[21] Juan Du, Daniel J. Dean, Yongmin Tan, Xiaohui Gu, and Ting Yu. "Scalable Distributed Service Integrity Attestation for Software-as-a-Service Clouds", in IEEE Transactions on Parallel and Distributed Systems, Vol. 25, No. 3, March 2014.

[22] Valentin Tudor, Magnus Almgren, and Marina Papatriantafilo. "Analysis of the Impact of Data Granularity on Privacy for the Smart Grid", in Proc. of the 12th Annual ACM Workshop on Privacy in the Electronic Society, Berlin, Germany, 2013.