A Resilient and Trustworthy Cloud and Outsourcing Security Framework for Power Grid Applications

Argonne National Laboratory (ANL)

Feng Qiu

Cybersecurity for Energy Delivery Systems Peer Review

November 6-8, 2018
Summary: A Resilient and Trustworthy Cloud and Outsourcing Security Framework for Power Grid Applications

Objective

- **Background**: Cloud computing provides powerful computational capacity, scalability, and high cost-effectiveness
- **Challenges**: Confidentiality of grid data; vulnerabilities in data transmission and cloud data storage; time criticality
- **Opportunity**: Build a trustworthy and secured cloud computing framework for power grid applications to facilitate cloud computing in power industry
- **Benefits**: Provide highly secured encryption framework for power system computing (on cloud or other outsourcing scenarios)

Schedule

- Started in August 2016, ends August 2021
- Key deliverables and dates met
  - Design of an attack-resilient framework, Y1 Q2
  - Deployment of SCED & SCUC on cloud, Y2 Q4
  - Capabilities to be transitioned to energy sector
    - Attack-resilient framework for power system applications on cloud computing and other outsourced platforms
    - Privacy-preserving methodologies and software packages for a set of power system applications

**Total Value of Award**: $1,500,000

**Funds Expended to Date**: %40

**Performer**: Argonne National Lab

**Partners**: University at Buffalo, Illinois Institute of Technology
Advancing the State of the Art (SOA)

- Cloud Computing
  - Powerful: Amazon EC2 96 vCPUs 345 GB memory
  - Scalable: Hundreds or even thousands of instances simultaneously
  - Cost-effective: $0.0016/hr (spot pricing)
  - Nearly half of all companies claim 31% to 60% of their IT systems are cloud-based
  - Global Smart Grid as a Service market expected to grow from $1.3 billion in 2016 to $6 billion in 2025 [“Smart Grid as a Service,” Navigant Research, 2016]

- Weak Cloud Security
  - Shared Security Responsibility Model
    - Secure only certain layers of infrastructure and software
    - Customer is ultimately responsible for how data are accessed/used
  - Data breaches on cloud
    - AWS, Microsoft, Apple, Yahoo . . .
    - Malware injection, side channel, wrapping, Spectre, and Meltdown (shared memory)

- Commonly Used Cloud Cybersecurity Methods
  - Communication encryption, data encryption
  - Cloud computing is completely vulnerable to insider attacks
  - Not suitable for power system computing

- Privacy-Preserving (PP) Methodologies
  - “Fake” problems solved on cloud; real data always on local
  - Data confidentiality is preserved even if data breach occurs
  - Ensuring correctness, optimality, and performance of solution
Advancing the State of the Art (SOA) (cont.)

- **A Holistic Security Framework for Cloud Computing**
  - Infrastructure security
  - Data confidentiality (privacy-preserving)
  - Application-specific encryption for higher security: Security-constrained economic dispatch (SCED), security-constrained unit commitment (SCUC), stochastic unit commitment (UC), etc.

- **Benefits to Cyber Resilience of Energy Delivery Systems**
  - Establish cybersecurity framework/methodologies for power system cloud computing
  - Pave the way (cybersecurity) to facilitate cloud computing application in power industry
Challenges to Success

Infrastructure Security

• High confidentiality of power grid data and insufficient cloud security
• Module-based cybersecurity system design for data transmission and storage

Data Integrity

• Power system computations completely vulnerable on cloud (leaking and manipulation)
• Set of encryption and validation methodologies ensure data confidentiality, accuracy, and consistency in computing

Time Criticality

• Applications must be completed in a timely manner to ensure continuous operation; time cost of encryption
• Highly efficient and effective privacy-preserving methods
Progress to Date

Major Accomplishments

• Diverse Industry Advisory Board
  • Xiaochuan Luo, ISO-NE; Jianzhong Tong, PJM
  • Alex Rudkevich, Newton Energy Group; Tobias Whitney, EPRI (Cyber Security for the Electric Sector)

• Important Milestones Accomplished (progress on track)
  • Design of an attack-resilient framework that comprehensively captures all common cyber and physical properties across power grid monitoring, protection, and control applications
  • Model of attacks against cloud-based power grid applications
  • Deployment of SCED and SCUC on GovCloud (AWS)
  • Initial results on privacy-preserving methods on SCED and SCUC

• Publications
  • “Cyberattacks Against Cloud-based Power System Applications,” ANL/ESD-18/16, Lemont, IL, Argonne National Laboratory.
  • “Privacy-preserving Transformations for Security Constrained Unit Commitment” (in preparation).
Collaboration/Technology Transfer

Plans to Transfer Technology/Knowledge to End User

• Reduce Technology Adaption Difficulty
  • Modular design for flexible implementation and deployment
  • Thorough test on publicly accessible clouds

• Stick to Industrial Needs
  • Select widely used power system applications to develop cloud security enhancement
  • Emphasize practicality and scalability (large-scale systems will be thoroughly tested)
  • Industry advisory board with various potential customers

• End-users Include but not Limited to:
  • System Operators: Directly implement on cloud services
  • Software as a service (SaaS): Entity can host and maintain the technology framework for a usage/service fee

• Testing and Demonstrate Plan
  • Demonstration to industry with realistic instances (PJM, etc.)
Next Steps for this Project

Approach for the Next Year or to the End of Project

• Sparse Transformation for SCUC
  • Sparse transformation for integer programming
  • Selectively secure certain data (e.g., topology) to achieve higher performance

• Distributed Cyber Security Framework
  • Enhanced security by distributing data and computations on multiple machines
  • Enhanced computational performance by parallel computing

• Security Enhancement for Stochastic UC on Cloud
  • One of the applications that can benefit most from cloud computing
  • Utilizing a large pool of computers on cloud

• Implementation and Test for Industrial Adaption
  • Scalability and technology transferability
Infrastructure Security Framework

System Framework of Resilient and Trustworthy Cloud and Outsourcing Framework

- Identity and Access Management
- Confidentiality evaluator
- Communication security and authentication
- Virtual firewall
- CSP components
- Data audit protocols
- Result verification schemes
Transformation-Based Privacy-Preserving

**Desired Security Definition**
- Assumption: Attackers know the model but not the data
- The number of values in this domain is infinite, or the number of values in this domain is so large that a brute-force attack is computationally infeasible.
- The range of the domain (the difference between the upper and lower bounds) is acceptable for the application.

**Transformations**
- Multiplying from left/right, scaling and perturbation, shifting

\[
\begin{align*}
\min \quad & c^T x \\
\text{s.t.} \quad & Mx \leq B \\
\quad & x \geq 0
\end{align*}
\]

**Perturbation/Scaling**
\[
\begin{align*}
\min \quad & c^T Q(Q^{-1}x + r) \\
M_1 Q(Q^{-1}x + r) &= b_1 + M_1 Q r \\
M_2 Q(Q^{-1}x + r) &\leq b_2 + M_2 Q r \\
(Q^{-1}x + r) &\geq r
\end{align*}
\]

\(Q\) a positive monomial matrix

\[
M' = \begin{pmatrix} M_1 Q & 0 \\ M_2 Q & A \end{pmatrix}, \quad b' = \begin{pmatrix} b_1 + M_1 Q r \\ b_2 + M_2 Q r \\ -S r \end{pmatrix}
\]

**Add slack variables**

\[
\begin{align*}
\min \quad & c_s^T z_s \\
\text{s.t.} \quad & M' z_s = b' \\
& z_s \geq 0
\end{align*}
\]

**Multiplying by random non-singular matrix**

\[
\begin{align*}
M'' &= P \cdot M' \\
b'' &= P \cdot b' \\
\min \quad & c_s^T z \\
\text{s.t.} \quad & M'' z_s = b'' \\
& z_s \geq 0
\end{align*}
\]

\(P\): non-singular matrix
Privacy-Preserving SCED

PPSCED–An Illustration (Heat maps indicate the no-zero coefficient density)

Comparing AWS Cloud with In-house HPC (ANL Blues)

<table>
<thead>
<tr>
<th>COMPUTING INFRASTRUCTURE CHARACTERISTICS</th>
<th>CPU</th>
<th>RAM</th>
<th>SSD</th>
<th>Intel Processor</th>
<th>$/h</th>
</tr>
</thead>
<tbody>
<tr>
<td>1) ANL Blues</td>
<td>16</td>
<td>64</td>
<td>✓</td>
<td>Xeon Nehalem</td>
<td>2.880</td>
</tr>
<tr>
<td>2) c4.2xlarge</td>
<td>8</td>
<td>16</td>
<td>✓</td>
<td>Xeon E5-2666v3</td>
<td>0.419</td>
</tr>
<tr>
<td>3) c4.4xlarge</td>
<td>16</td>
<td>30</td>
<td>✓</td>
<td>Xeon E5-2666v3</td>
<td>0.838</td>
</tr>
<tr>
<td>4) c4.8xlarge</td>
<td>36</td>
<td>60</td>
<td>✓</td>
<td>Xeon E5-2666v3</td>
<td>1.675</td>
</tr>
<tr>
<td>5) m4.16xlarge</td>
<td>64</td>
<td>256</td>
<td>✓</td>
<td>Xeon E5-2686v4</td>
<td>3.830</td>
</tr>
</tbody>
</table>

- Simulating SCED on 2383-bus Polish system, run every 5 minutes, compare performance and costs
- Shuffling and scaling
- Cost effective: 77-85% saving over ANL Blues
- Cloud provide a variety of performance options
Privacy-Preserving Transformation for SCUC

Performance vs. Security

- SCUC: Computational performance of integer programming is very sensitive to constraint matrix density

A Shuffling and Scaling Method

\[
\begin{align*}
\text{minimize} & \quad \sum_{i \in I} \sum_{g \in G} \left[ c^U_{ik} y_{g,t} + c^D_{ik} z_{g,t} + c^{\min}_{ik} x_{g,t} + \sum_{k \in K} c^k_{ik} p^k_{g,t} \right] \\
\text{subject to} & \quad p_{g,t} = p_{g,t}^{\min} x_{g,t} + \sum_{k \in K} p^k_{g,t} \\
& \quad p^k_{g,t} \leq p_{g,t}^{\min} x_{g,t} \\
& \quad p_{g,t} \leq p_{g,t-1} + R^U_{g,t} \\
& \quad p_{g,t} \geq p_{g,t-1} - R^D_{g,t} \\
& \quad \sum_{g \in G} p_{g,t} = D_t \\
& \quad x_{g,t} - x_{g,t-1} = y_{g,t} - z_{g,t} \\
& \quad -F_t - \sum_{b \in B} \sum_{g \in G} \delta_b^{l} p_{b,t} \leq F_t + \sum_{b \in B} \delta_b^{l} p_{b,t} \\
& \quad p \geq 0 \\
& \quad x_{g,t}, y_{g,t}, z_{g,t} \in \{0, 1\}
\end{align*}
\]

\[
\begin{align*}
\text{minimize} & \quad \sum_{i \in I} \sum_{g \in G} \left[ \gamma c^U_{ik} y_{g,t} + \gamma c^D_{ik} z_{g,t} + \gamma c^{\min}_{ik} x_{g,t} + \sum_{k \in K} \gamma c^k_{ik} D_t p^k_{g,t} \right] \\
\text{subject to} & \quad p_{g,t} = \frac{p_{g,t}^{\min}}{D_t} x_{g,t} + \sum_{k \in K} p^k_{g,t} \\
& \quad p^k_{g,t} \leq \frac{p_{g,t}^{\min}}{D_t} x_{g,t} \\
& \quad p_{g,t} \leq \frac{D_{t-1}}{D_t} p_{g,t-1} + \frac{R^U_{g,t}}{D_t} \\
& \quad p_{g,t} \geq \frac{D_{t-1}}{D_t} p_{g,t-1} - \frac{R^D_{g,t}}{D_t} \\
& \quad \sum_{g \in G} p_{g,t} = 1 \\
& \quad x_{g,t} - x_{g,t-1} = y_{g,t} - z_{g,t} \\
& \quad -\frac{\alpha_t F_t}{D_t} - \sum_{b \in B} \frac{\alpha_t}{D_t} \delta_b^{l} p_{b,t} \leq \sum_{b \in B} \frac{\alpha_t}{D_t} \delta_b^{l} p_{b,t} \leq \sum_{b \in B} \frac{\alpha_t}{D_t} \delta_b^{l} p_{b,t} \\
& \quad p \geq 0 \\
& \quad x_{g,t}, y_{g,t}, z_{g,t} \in \{0, 1\}
\end{align*}
\]
Privacy-Preserving Transformation for SCUC

Security

- Partially secured (absolute values protected but not relative values)
  - Start-up, shutdown, production costs, generation capacities, ramping rates, demands
- Perfectly secured
  - Network topology (PTDF matrix) and thermal limits
- Implementation
  - Julia 0.6.4, JuMP 0.18.4, CPLEX 12.8.0
  - GovCloud, SSH

Performance

<table>
<thead>
<tr>
<th>instance</th>
<th>host</th>
<th>t-key</th>
<th>t-enc</th>
<th>t-solve</th>
<th>t-comm</th>
<th>t-total</th>
<th>obj</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case1951</td>
<td>1xlarge</td>
<td>0.08</td>
<td>1.32</td>
<td>131.82</td>
<td>1.32</td>
<td>134.54</td>
<td>52660765</td>
</tr>
<tr>
<td></td>
<td>2xlarge</td>
<td>0.07</td>
<td>1.26</td>
<td>171.42</td>
<td>1.29</td>
<td>174.04</td>
<td>52644059</td>
</tr>
<tr>
<td></td>
<td>4xlarge</td>
<td>0.08</td>
<td>1.33</td>
<td>128.02</td>
<td>1.32</td>
<td>130.74</td>
<td>52676234</td>
</tr>
<tr>
<td></td>
<td>notebook</td>
<td>0.08</td>
<td>1.10</td>
<td>147.73</td>
<td>1.78</td>
<td>150.69</td>
<td>52676234</td>
</tr>
<tr>
<td>Case2848</td>
<td>1xlarge</td>
<td>0.36</td>
<td>1.44</td>
<td>352.91</td>
<td>1.36</td>
<td>356.06</td>
<td>53631900</td>
</tr>
<tr>
<td></td>
<td>2xlarge</td>
<td>0.36</td>
<td>1.37</td>
<td>320.33</td>
<td>1.35</td>
<td>323.41</td>
<td>53634704</td>
</tr>
<tr>
<td></td>
<td>4xlarge</td>
<td>0.37</td>
<td>1.45</td>
<td>325.05</td>
<td>1.3</td>
<td>328.16</td>
<td>53634044</td>
</tr>
<tr>
<td></td>
<td>notebook</td>
<td>0.39</td>
<td>1.14</td>
<td>282.3</td>
<td>2.21</td>
<td>286.03</td>
<td>53630267</td>
</tr>
<tr>
<td>Case3375</td>
<td>1xlarge</td>
<td>0.12</td>
<td>2.75</td>
<td>592.33</td>
<td>1.97</td>
<td>597.17</td>
<td>46532888</td>
</tr>
<tr>
<td></td>
<td>2xlarge</td>
<td>0.11</td>
<td>2.71</td>
<td>483.33</td>
<td>1.9</td>
<td>488.04</td>
<td>46531362</td>
</tr>
<tr>
<td></td>
<td>4xlarge</td>
<td>0.12</td>
<td>2.77</td>
<td>511.61</td>
<td>1.85</td>
<td>516.35</td>
<td>46525589</td>
</tr>
</tbody>
</table>
|          | notebook | 0.13 | 2.13  | 660.2   | 2.93   | 665.38  | 46525413   

Constraint Matrix after PP Transformation
Advantages of distributed security framework

- Scalability by parallel computing
- Stronger security framework

Distributed security workflow

1) partition the grid application into a set of smaller sub-problems and a master problem
2) Encrypt each sub-problem (with PP) and send to a cloud server; master problem with critical information kept on local
3) Solve each encrypted sub-problem and pass back solution
4) Solve master problem and send updates to sub-problems
5) Iterate until convergence criteria met

Security features

- Hard to track: each time use different partitions, solved on different servers
- Hard to recover valuable information: distributed information; encrypted independently
- Security at multiple levels
Challenges

• Decomposable structure and sparsity
• Computational performance: convergence, solution time, parallel implementation

Novel decompositions for network constraints

• Reformulations of network constraints that have been used for decades in power engineering
• Sparse and decomposable structure
• Strong computational performance
• Working on distributed computing with security enhancement

Instance:
• Simplified version of Polish test system: 3375 buses, 596 units, 4076 branches and 9 zones

Results:
• 64% reduction in non-zeros
• 2.4x faster running time

<table>
<thead>
<tr>
<th>Matrix</th>
<th>Reduced MIP nz</th>
<th>Running Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original Form.</td>
<td>2,924,357</td>
<td>430 s</td>
</tr>
<tr>
<td>Decomposable</td>
<td>1,029,175</td>
<td>178 s</td>
</tr>
</tbody>
</table>
Thank you!

Feng Qiu, PhD
Principle Computational Scientist
Energy Systems Division
Argonne National Laboratory
9700 S. Cass Ave, Lemont, IL 60439
fqiu@anl.gov