A Privacy Preserving Approach to Energy Theft Detection in Smart Grids

Christopher Richardson and Nicholas Race
School of Computing & Communications
Lancaster University, UK
c.richardson@lancaster.ac.uk, n.race@lancaster.ac.uk

Paul Smith
Digital Safety & Security Department
AIT Austrian Institute of Technology
Vienna, Austria
paul.smith@ait.ac.at
Aim to detect tampering of smart meters for financial gain

Based on a diagram by Tom Chevalier
Privacy Issue

It is important to be able to detect energy theft, while still preserving the privacy of customers.

Fine grained data can leak sensitive information.
Smart Meter Tampering
Distribution of solar nodes

The data was obtained from the Sheffield Solar Group at the University of Sheffield: http://www.microgen-database.org.uk/
Normalised Solar Power Electricity Production
Trusted Third-Party (e.g., Energy Supplier)

- Create geospatial regions of PV installations
- Generate public-private key-pair per geospatial region
- Send the public key to all the BEAs in a region
- For each region, generate all combinations of pairs of BEAs

For each BEA pair, denote one to be Bob and the other Alice

Instruct all Bobs to calculate the Euclidean distance between their normalised generation data and its designated Alice, for a day

For all results from Bobs, decrypt the Euclidean distances

Per geospatial region, cluster distances using DBSCAN to identify outliers

Bob (BEA)

- Send public key to Alice and request encrypted normalised generation measurements
- Calculate Euclidean distance using homomorphic computation and return encrypted result

Alice (BEA)

- Calculate normalised power generation measurements, encrypt and send to Bob
Paillier Cryptosystem

- Partially homomorphic

- Supports addition operation

- Also supports multiplication, via use of a plaintext and ciphertext
Euclidean Distance

Sum of the difference between two houses solar panel outputs, across a day
Euclidean Distance

\[ d(q_1, p_1) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2} \]

Equivalent to the below (necessary as we can’t perform exponentiation on ciphertexts where we don’t know the plaintext with Paillier):

\[
\begin{align*}
A &= \sum_{i=1}^{n} q_i^2 \\
B &= \sum_{i=1}^{n} p_i^2 \\
C &= -\sum_{i=1}^{n} 2q_ip_i
\end{align*}
\]

\[ d(q_1, p_1) = \sqrt{A + B + C} \]

Experimental Platform

In ‘Software Architecture for a Smart Grids Test Facility - IT Implementation for an Emulated Low Voltage Smart Grid’ BEA is realised through the use an embedded industrial PC, such as the Siemens Nanobox PC SIMATIC IPC227D 19

Raspberry Pi & Mininet
Initial Results – F1 Score

\[ F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \]

\[ \text{precision} = \frac{tp}{tp + fp} \]

\[ \text{recall} = \frac{tp}{tp + fn} \]
Conclusion

- Privacy issues highlighted

- Preliminary tests of our system on two testbeds (Raspberry Pis & Mininet)

- Initial accuracy of results presented
Future Work

- Experimentation with differing geospatial sizes
- Filtering of data prior to euclidean distance
- Using similar techniques for attack detection in other systems
Any Questions?

Also feel free to e-mail me at
c.richardson@lancaster.ac.uk