Privacy-friendly Aggregation for the Smart-grid

presenter: Le Chen

Nanyang Technological University

lechen0213@gmail.com

September 28, 2013

Overview

- Introduction
- Basic Protocols
- Concrete Protocols
- Analysis
- Conclusion & Discussion

Reference

- Klaus Kursawe, George Danezis, and Markulf Kohlweiss. *Privacy-friendly Aggregation for the Smart-grid*. Privacy Enhancing Technologies (PETS) 2011, Waterloo, ON, Canada, July 27-29, 2011. LNCS 6794, 2011, pp 175-191.

Motivation

- Aggregates of consumption across different populations are used for *leakage detection, fraud detection, forecasting, tuning production to demand, settling the cost of production across electricity suppliers*, etc.
- Aggregation protocols will also be used to detect leakages in other utilities, e.g., water (which is a big issue in desert countries) and gas (where a leakage poses a safety problem).
- By aggregation, the communication overhead and storage needed can be dramatically decreased.

Privacy in Smart Metering

- The high frequency suggested (i.e., about 15 minutes reading interval) for electricity usage metering totally exposes one's behavior privacy.
- An important aspect in privacy preserving metering protocols is to take into account the rather limited resources on such meters, both in terms of bandwidth and in terms of computation.

Basic Ideas

- The protocols we proposed is relying on masking the meter consumptions $c_{t,j}$ output by meter j for a reading interval t, in such a way that an adversary cannot recover individual readings.
- The sum of the masking values across meters sums to a known value (e.g. 0).
- To prevent linking masked values, the masks are recomputed for every measurement.

Aggregation Protocols

- Metered homes use masking values $x_{t,j}$ to output blinded values $c_{t,j} + x_{t,j}$.
- After the masking values have canceled each other out ($\sum_j x_{t,j} = 0$), the result of the protocol is $\sum_j c_{t,j}$.
- Note that this is a kind of protocols.

Comparison Protocols

- Homes output $g^{c_{t,j}+x_j}$ and the result of the protocol is $g^{\sum_j c_{t,j}}$.
- They require that the aggregator already knows the (approximate) sum of the values she is aggregating (through a feeder meter), and needs to determine whether her sum is sufficiently close to the aggregate obtained from home meters.
- One advantage is that in contrast to aggregation protocols, no fresh $x_{t,j}$ are needed, i.e. x_j is fixed.
- Note that this is a kind of protocols, too.

Comparison Protocols

The basic comparison protocol.

- $H : \{0,1\}^* \to G$, $g_t = H(t)$.
- Pre-installed x_j, s.t. $\sum x_j = 0$.
- Home j : $g_{t,j} = g_t^{c_{t,j}+x_j}$.
- Aggregator: $g_a = \prod_j g_{t,j} = g_t^{\sum_j c_{t,j}}$.
  c_a: approximate, brute force $g_t^{c_a}, g_t^{c_a-1}, g_t^{c_a+1}, \ldots$

Interactive Protocol

Our first protocol uses simple additive secret sharing.

- For each round t, choose p leaders from meters.
- Each home j generates p random numbers for each leader: $s_{j,1}, \ldots, s_{j,p}$.

$$
\begin{array}{cc}
 & \begin{array}{ccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 \end{array} \\
\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} &
\begin{pmatrix}
 & s_{2,1} & s_{3,1} & s_{4,1} & s_{5,1} & s_{6,1} & s_{7,1} \\
s_{1,2} & & s_{3,2} & s_{4,2} & s_{5,2} & s_{6,2} & s_{7,2} \\
s_{1,3} & s_{2,3} & & s_{4,3} & s_{5,3} & s_{6,3} & s_{7,3} \\
s_{1,4} & s_{2,4} & s_{3,4} & & s_{5,4} & s_{6,4} & s_{7,4}
\end{pmatrix}
\end{array}
$$

Interactive Protocol

Our first protocol uses simple additive secret sharing.

- For each round t, choose p leaders.
- Each home j generates p random numbers for each leader: $s_{j,1}, \ldots, s_{j,p}$.
- Each leader k generates $s_{k,k}$ s.t. $\sum_{i=1}^{n} s_{i,k} = 0$.
- Let $s_j = \sum_i s_{j,i}$.

$$
\begin{array}{c c c}
 & \begin{array}{ccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 \end{array} & \\
\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{pmatrix} \color{red}{s_{1,1}} & s_{2,1} & s_{3,1} & s_{4,1} & s_{5,1} & s_{6,1} & s_{7,1} \\ s_{1,2} & \color{red}{s_{2,2}} & s_{3,2} & s_{4,2} & s_{5,2} & s_{6,2} & s_{7,2} \\ s_{1,3} & s_{2,3} & \color{red}{s_{3,3}} & s_{4,3} & s_{5,3} & s_{6,3} & s_{7,3} \\ s_{1,4} & s_{2,4} & s_{3,4} & \color{red}{s_{4,4}} & s_{5,4} & s_{6,4} & s_{7,4} \end{pmatrix} & \begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \end{array} \\
 & \begin{array}{ccccccc} s_1 & s_2 & s_3 & s_4 & s_5 & s_6 & s_7 \end{array} &
\end{array}
$$

Interactive Protocol

For the aggregation protocol:

- let $x_{t,j} = s_j$.
- To update the masking values, the above steps are repeated with a different set of leaders for each round t.
- $b_{t,j} = c_{t,j} + x_{t,j} \mod 2^{32}$, thus $\sum_j c_{t,j} = \sum_j b_{t,j} \mod 2^{32}$.

For the comparison protocol:

- The interactive protocol can also be used in combination with the basic comparison protocol by setting $x_j = s_j$, removing the need for updating shares.

Diffie-Hellman Key-Exchange Based Protocol

- For each round t, $g_t = H(t)$.
- Each home j computes a round specific public key $\mathrm{Pub}_{t,j} = g_t^{X_j}$, and distributes it to all.
- Each home collects $\mathrm{Pub}_{t,1}$, ..., $\mathrm{Pub}_{t,n}$, and computes

$$g_t^{x_j} = \prod_{k \neq j} \mathrm{Pub}_{t,k}^{(-1)^{I(k<j)} X_j},$$

where $I(k<j) = 1$ while $k < j$, 0 otherwise. And we have

$$\sum_j x_j = \sum_j \sum_{k \neq j} (-1)^{I(k<j)} X_k \cdot X_j = 0$$

Diffie-Hellman Key-Exchange Based Protocol

$$\sum_j x_j = \sum_j \sum_{k \neq j} (-1)^{I(k<j)} X_k \cdot X_j = 0.$$

$$\begin{array}{c c c}
 & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} & k \\
\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{pmatrix} null & X_2X_1 & X_3X_1 & X_4X_1 \\ -X_1X_2 & null & X_3X_2 & X_4X_2 \\ -X_1X_3 & -X_2X_3 & null & X_4X_3 \\ -X_1X_4 & -X_2X_4 & -X_3X_4 & null \end{pmatrix} & \begin{array}{c} x_1 \\ x_2 \\ x_3 \\ x_4 \end{array} \\
j & &
\end{array}$$

Diffie-Hellman Key-Exchange Based Protocol

$$\sum_j x_j = \sum_j \sum_{k \neq j} (-1)^{I(k<j)} X_k \cdot X_j = 0.$$

$$\begin{array}{cc c}
 & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} & k \\
\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{pmatrix} null & X_2X_1 & X_3X_1 & X_4X_1 \\ -X_1X_2 & null & X_3X_2 & X_4X_2 \\ -X_1X_3 & -X_2X_3 & null & X_4X_3 \\ -X_1X_4 & -X_2X_4 & -X_3X_4 & null \end{pmatrix} & \begin{array}{c} x_1 \\ x_2 \\ x_3 \\ x_4 \end{array} \\
j & &
\end{array}$$

$$R_{(i,t)} = N + \sum_{j=1, i \neq j}^{k} r_{(i \to j, t)} - \sum_{j=1, i \neq j}^{k} r_{(j \to i, t)}.$$

Diffie-Hellman Key-Exchange Based Protocol

- No aggregation protocol. Because x_j cannot be known or recovered by any other meters.
- For the comparison protocol, each meter computes:

$$g_{t,j} = g_t^{c_{t,j}} \cdot g_t^{x_j} = g_t^{c_{t,j}+x_j}$$

Diffie-Hellman and Bilinear-map Based Protocol

- $e(\mathbb{G}_1, \mathbb{G}_2) \to \mathbb{G}_T$, $H : \{0,1\}^* \to \mathbb{G}_2$.
- Each home j computes fixed public key $\mathrm{Pub}_j = \hat{g}_0^{X_j}$, where $\hat{g}_0$ is a generator of $\mathbb{G}_1$.
- In round t, homes compute $\hat{g}_t = H(t)$ and let $g_t = e(\hat{g}_0, \hat{g}_t)$. Homes can now compute $g_t^{x_j}$ as

$$g_t^{x_j} = \Big(\prod_{k \neq j} e(\mathrm{Pub}_k, \hat{g}_t)\Big)^{(-1)^{I(k<j)} X_j},$$

where $I(k<j) = 1$ while $k < j$, 0 otherwise. And we have

$$\sum_j x_j = \sum_j \sum_{k \neq j} (-1)^{I(k<j)} X_k \cdot X_j = 0$$

Diffie-Hellman and Bilinear-map Based Protocol

- No aggregation protocol. Because x_j cannot be known or recovered by any other meters.
- For the comparison protocol, each meter computes:

$$g_{t,j} = g_t^{c_{t,j}} \cdot g_t^{x_j} = g_t^{c_{t,j}+x_j}$$

Low-overhead Protocol

- Similar as the Bilinear map based scheme, we assume that all meters have a fixed public key $\mathrm{Pub}_j = g^{X_j}$.
- Each home j computes a set of shared keys, as: $K_{j,k} = H(\mathrm{Pub}_k^{X_j})$.
- In round t of masking value generation, each meter j outputs:

$$x_{t,j} = \sum_{k \neq j} (-1)^{I(k<j)} H(K_{j,k} || t)$$

Low-overhead Protocol

- For the aggregation protocol, only 32 bits of $x_{t,j}$ are needed. Let $b_{t,j} = c_{t,j} + x_{t,j} \mod 2^{32}$, we have $\sum_j c_{t,j} = \sum_j b_{t,j} \mod 2^{32}$.
- For the comparison protocol, set $x_j = x_{t',j}$ for a fixed t', then we have

$$g_{t,j} = g^{c_{t,j}} \cdot g^{x_j} = g^{c_{t,j}+x_j}$$

Privacy

- If all participants are honest-but-curious and do not collude, the privacy is maintained.
- In case of collusion, the DH based protocol, the bilinear maps based protocol, and the low-overhead protocol ensure that the anonymity set within which meter readings are aggregated includes all the non colluding meter readings.
- The interactive protocol has a similar property for any number of colluding nodes that does not include all leaders. If all leaders collude all privacy is lost.

Converting an Comparison Protocol back into an Aggregation Protocol

- In some scenarios, there is no feeder meter that provides the approximate sum.
- A typical smart meter reading is a four byte value. If we assume up to 250 devices in one group, that would give us a 40 bit value for the aggregated reading.
- In most cases, the aggregator has a fairly good idea on the rough total consumption.
- A normal computer can brute-force the sum in a reasonable short time.

Conclusion & Discussion

- This paper proposes several privacy-friendly aggregation schemes relying on masking the meter consumptions.
- Based on the feature that meter readings are small and predicable, brute-force computation can be used after the relative large masking values are canceled.
- Discussion? (Differences between the scheme I presented last week?)