A New Communication-Efficient Privacy-Preserving Range Query Scheme in Fog-Enhanced IoT

Rongxing Lu

Contact Email: rlu1@unb.ca

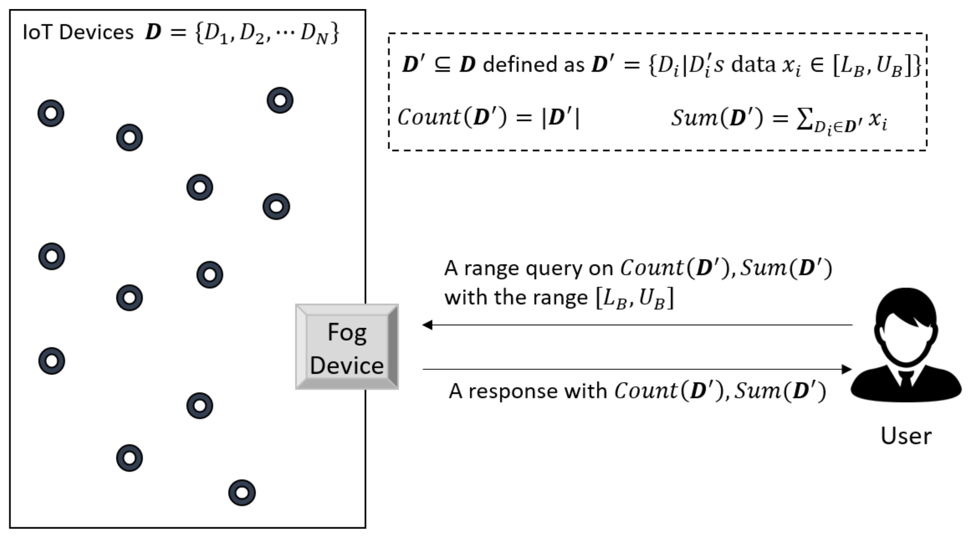
Canadian Institute for Cybersecurity (CIC), Faculty of Computer Science, University of New Brunswick (UNB)

ABSTRACT

Fog-enhanced Internet of Things (IoT) has received considerable attention in recent years, as fog devices deployed at the network edge can not only improve the performance of IoT applications, but also enhance the security and privacy of IoT. In this work, we present a new communication-efficient privacy-preserving range query scheme in Fog-enhanced IoT. With the proposed scheme, both the query range and individual IoT device's data can be privacy-preserved by using BGN homomorphic encryption technique. In addition, the proposed scheme employs a range query expression, decomposition, and composition technique to reorganize the range query, which can achieve $O(\sqrt{n})$ communication efficiency. Extensive experiments are conducted, and the results indicate that the proposed scheme is efficient in terms of communication overhead.

System Model

- **IoT devices $\boldsymbol{D} = \{D_1, \cdots, D_N\}$:** Each $D_i \in \boldsymbol{D}$ is equipped with sensing and periodically sends the sensed data $w_i \in [1, n]$ to the fog device.
- **Fog device:** The fog device can process the data collected from IoT devices and handle the range query request from the query user.
- **Query user:** A query user will launch range queries to the fog device and gain the desirable result from the fog device.



Design Goals

- The proposed scheme should be privacy-preserving, i.e., the query range $[L_B, U_B]$ and the elements of subset D' should be privacy-preserving.
- The proposed scheme should be communication efficient, i.e., achieving $\sqrt{n}$ query communication efficiency.

A. Range Query Expression, Decomposition, and Composition

(1) Range Query Expression

A range query $[L_B, U_B]$ $(1 \le L_B \le U_B \le n)$ is first represented to an array $A[1..n]$ as shown in Fig. 1 and then reorganized into an $m \times m$ $(n = m^2)$ matrix R as

$$R(i,j) = \begin{cases} 1 & L_B \le k = (i-1) \times m + j \le U_B \\ 0 & otherwise \end{cases}$$

(2) Range Query Decomposition

RoI is a row in which not all elements are 0s or 1s;

BoI is a set of continuous rows in which all elements are 1s; Then, matrix R can be decomposed as the following steps.

Step-1: Break down R into three matrices R_1, R_2 and R_3 such that $R = R_1 \vee R_2 \vee R_3$, where R_1 and R_3 include at most one RoI respectively, and R_2 includes at most one BoI.

Step-2: Decompose R_w into two matrices $R_{1-X_w Y_w}$ and $R_{X'_w Y'_w}$ such that $R_w = R_{1-X_w Y_w} \wedge R_{X'_w Y'_w}$ for $w = 1,2,3$.

- Generate $R_{1-X_w Y_w}$ matrix: Set $X_w = (x_{w1}, x_{w2}, \cdots, x_{wm})$ with the row rule: if the i-th row in R_w are all 1s, set $x_{wi} = 0$ and set $x_{wi} = 1$ otherwise. Set $Y_w = (y_{w1}, y_{w2}, \cdots, y_{wm})$ with the column rule: if RoI (BoI) in R_w has an element 1 in column j, set $y_{wj} = 0$; and set $y_{wj} = 1$ otherwise. Then,
$$R_{1-X_w Y_w}(i,j) = 1 - x_{wi} y_{wj}$$
- Generate $R_{X'_w Y'_w}$ matrix: Set $X'_w = (x'_{w1}, x'_{w2}, \cdots, x'_{wm})$ with the row rule: if RoI (BoI) in R_w has an element 1 in row i, set $x'_{wi} = 1$; and set $x'_{wi} = 0$ otherwise. Set $Y'_w = (y'_{w1}, y'_{w2}, \cdots, y'_{wm}) = (1, 1, \cdots, 1)$. Then,
$$R_{X'_w Y'_w}(i,j) = x'_{wi} y'_{wj}$$

(3) Range Query Composition

The matrix R can be recovered by twelve vectors $(X_1, Y_1, X'_1, Y'_1, X_2, Y_2, X'_2, Y'_2, X_3, Y_3, X'_3, Y'_3)$ as

$$\begin{aligned} R(i,j) &= R_1(i,j) \vee R_2(i,j) \vee R_3(i,j) \\ &= \vee_{w=1}^{3} (R_{1-X_w Y_w}(i,j) \wedge R_{X'_w Y'_w}(i,j)) \\ &= \vee_{w=1}^{3} ((1 - x_{wi} y_{wj}) \wedge x'_{wi} y'_{wj}) \\ &= \textstyle\sum_{w=1}^{3} ((1 - x_{wi} y_{wj}) \cdot x'_{wi} y'_{wj}) \end{aligned}$$

Since all elements in vectors $(X_1, X_3, Y'_1, Y'_2, Y'_3)$ are 1s and all elements in the Y_2 are all 0s when R_2 includes one BoI. Then, $R(i,j) = \sum_{w=1}^{3} ((1 - x_{wi} y_{wj}) \cdot x'_{wi} y'_{wj})$

$$= (1 - y_{1j}) \cdot x'_{1i} + x'_{2i} + (1 - y_{3j}) \cdot x'_{3i}$$

$$= \begin{cases} 1 & within\ the\ query\ range \\ 0 & otherwise \end{cases}$$

Thus, the matrix R related to query can be recovered by $(Y_1, X'_1, X'_2, Y_3, X'_3)$ with size $O(5m) = O(\sqrt{n})$.

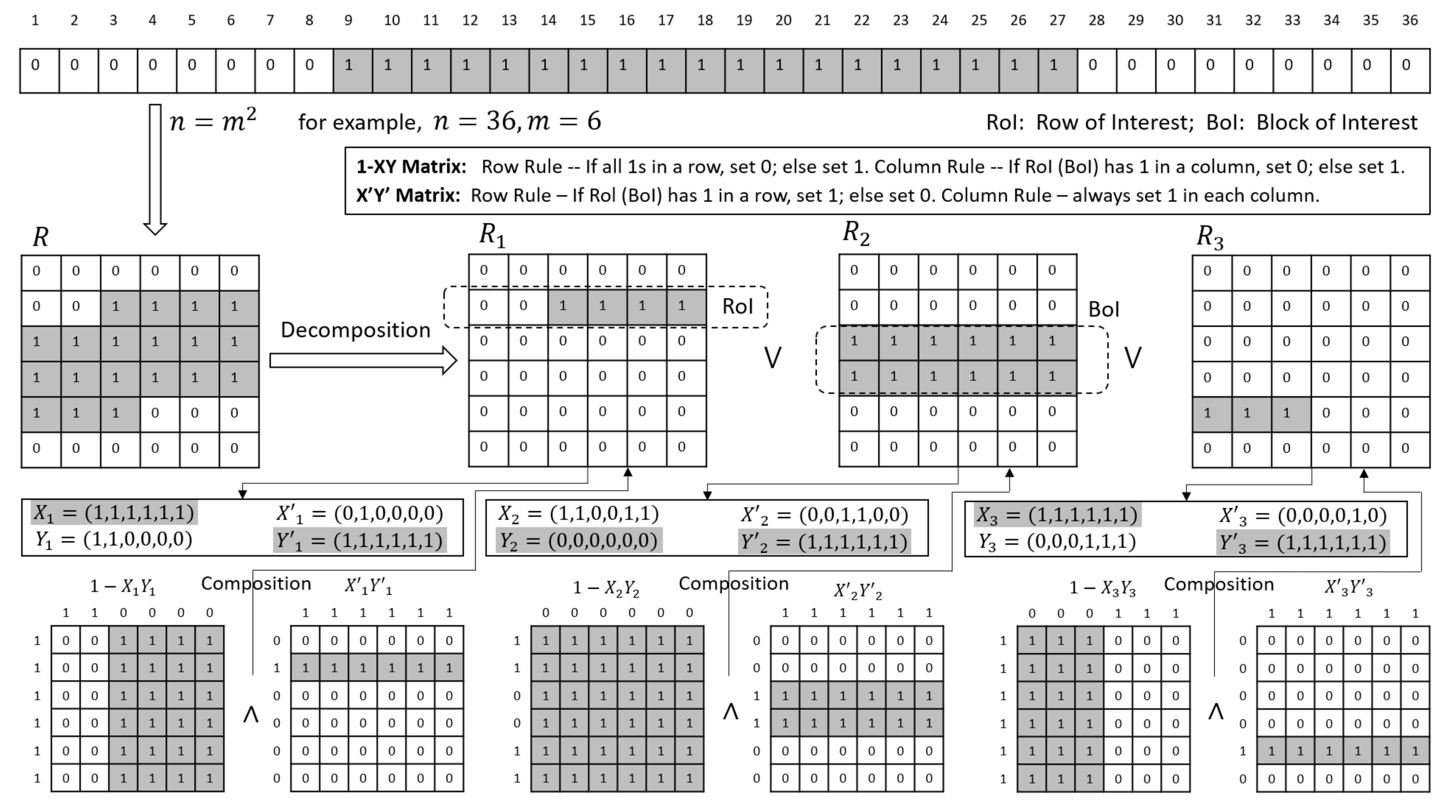


Fig. 1. An example for range query expression, decomposition, and composition

B. The proposed privacy-preserving range query scheme in fog-enhanced IoT

(1) Query User Key Generation: the query user generates the BGN public key pk and private key sk.

(2) Range Query Generation at Query User

- Represent the range query to a matrix R and apply the decomposition rules to prepare vectors $(Y_1, X'_1, X'_2, Y_3, X'_3)$. Then, the query user computes
$$\overline{Y_1} = (\overline{y_{11}} = 1 - y_{11}, \overline{y_{12}} = 1 - y_{12}, \cdots, \overline{y_{1m}} = 1 - y_{1m},)$$
$$\overline{Y_3} = (\overline{y_{31}} = 1 - y_{31}, \overline{y_{32}} = 1 - y_{32}, \cdots, \overline{y_{3m}} = 1 - y_{3m},)$$
Since $\mathrm{R(i,j)} = (1 - y_{1j}) \cdot x'_{1i} + x'_{2i} + (1 - y_{3j}) \cdot x'_{3i}$
$$= \overline{y_{1j}} \cdot x'_{1i} + x'_{2i} + \overline{y_{3j}} \cdot x'_{3i}$$
Then, R can be represented by $(\overline{Y_1}, X'_1, X'_2, \overline{Y_3}, X'_3)$.
- Use BGN to encrypt these vectors as $(E(\overline{Y_1}), E(X'_1), E(X'_2), E(\overline{Y_3}), E(X'_3))$.
- Send $(E(\overline{Y_1}), E(X'_1), E(X'_2), E(\overline{Y_3}), E(X'_3))$ as a query to all IoT devices via the fog device.

(3) Query Response at IoT Device

Each D_k with sensed data w_k performs the following steps.

Step-1: D_k converts the sensed data w_k into (i,j) such that
$$w_k = (i-1) \times m + j$$

Step-2: D_k picks up $E(\overline{y_{1j}}), E(x'_{1i}), E(x'_{2i}), E(\overline{y_{3j}}), E(x'_{3i})$, and chooses two random numbers r_{k_1} and r_{k_2}. Then, it computes

$$\begin{aligned} c_k &= e(E(\overline{y_{1j}}), E(x'_{1i})) \cdot e(E(x'_{2i}), g) \cdot e(E(\overline{y_{3j}}), E(x'_{3i})) \cdot e(g,h)^{r_{k_1}} \\ &= E_T(\overline{y_{1j}} x'_{1i} + x'_{2i} + \overline{y_{3j}} x'_{3i}) = E_T(R_k(i,j)) \end{aligned}$$

$$s_k = c_k^{w_k} \cdot e(g,h)^{r_{k_2}} = E_T(R_k(i,j))^{w_k} \cdot e(g,h)^{r_{k_2}} = E_T(R_k(i,j) \cdot w_k)$$

Step-3: D_k forwards (c_k, s_k) to the fog device.

(4) Response Aggregation at Fog Device

After receiving all (c_k, s_k) from all $D_k \in \boldsymbol{D}$, the fog device computes

$$C = \textstyle\prod_{D_k \in \boldsymbol{D}} c_k = E_T(\sum_{D_k \in \boldsymbol{D}} R_k(i,j))$$

$$S = \textstyle\prod_{D_k \in \boldsymbol{D}} s_k = E_T(\sum_{D_k \in \boldsymbol{D}} R_k(i,j) \cdot w_k)$$

and returns (C, S) as the response to the query user.

(5) Response Recovery at Query User

On receiving (C, S), the query user uses the private key sk to recover the query results as

$$C \xrightarrow{dec} Count(D') = |D'| = \sum_{D_k \in \boldsymbol{D}} R_k(i,j)$$

$$S \xrightarrow{dec} Sum(D') = \sum_{D_i \in D'} w_i = \sum_{D_k \in \boldsymbol{D}} R_k(i,j) \cdot w_k$$

Communication Overhead Analysis

The query user just sends 5 encrypted m-dimensional vectors to the fog device. Therefore, the communication overhead of the query user is $O(5m) = O(\sqrt{n})$.

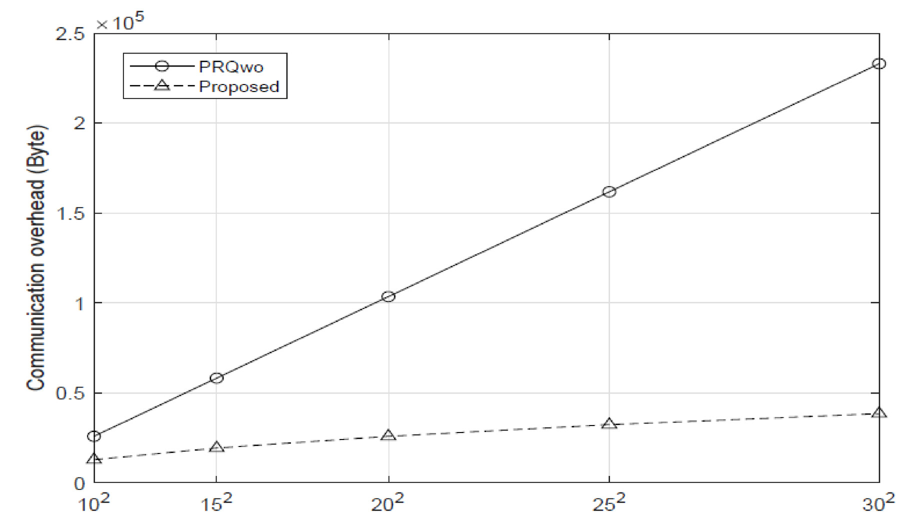


Fig. 2. Communication overhead comparisons between the proposed scheme and traditional scheme

* This work has been accepted by IEEE Internet of Things Journal in 2018.