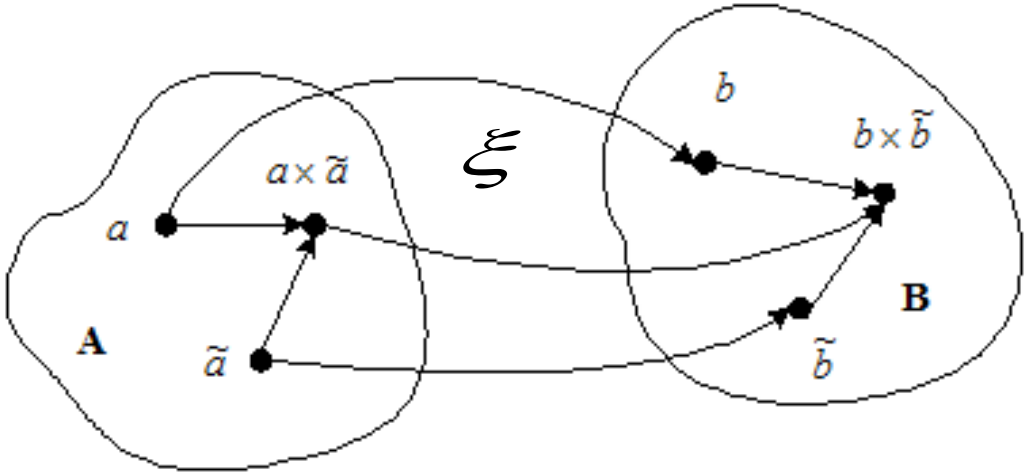
Гомоморфное шифрование

1. Понятие гомоморфного шифрования.
2. Криптосистема Пэйе.
3. Применение гомоморфного шифрование в протоколе скрытного определения «точек интереса».

Гомоморфизм



Пусть А и В группы и пусть ξ операция, отображающая элементы группы А в элементы группы В.

Отображение ξ(А→В) называется гомоморфным, если произведению (сложению) элементов в А, соответствует произведение (сложение) их отображений в В, т.е.

$$\xi(a \times \breve{a}) = \xi(в) \times \xi(\widetilde{в}).$$

$$\xi(a + \breve{a}) = \xi(в) + \xi(\widetilde{в}).$$

Гомоморфизм шифрования

Под гомоморфным шифрованием понимается криптографический примитив, представляющий собой функцию шифрования, удовлетворяющую дополнительному требованию гомоморфности относительно каких-либо алгебраических операций над открытыми сообщениями.

Пусть E(k,m) – функция шифрования, где m –открытое сообщение, k- ключ шифрования.

Функция шифрования E гомоморфна относительно операции ξ_1, над открытыми сообщениями, если существует операция ξ_2 над криптограммами такая, что из криптограммы $(E(k,m_1)\ \xi_2\ E(k,m_2))$ при дешифровании извлекается открытое сообщение $m1\xi_1m2$ то есть

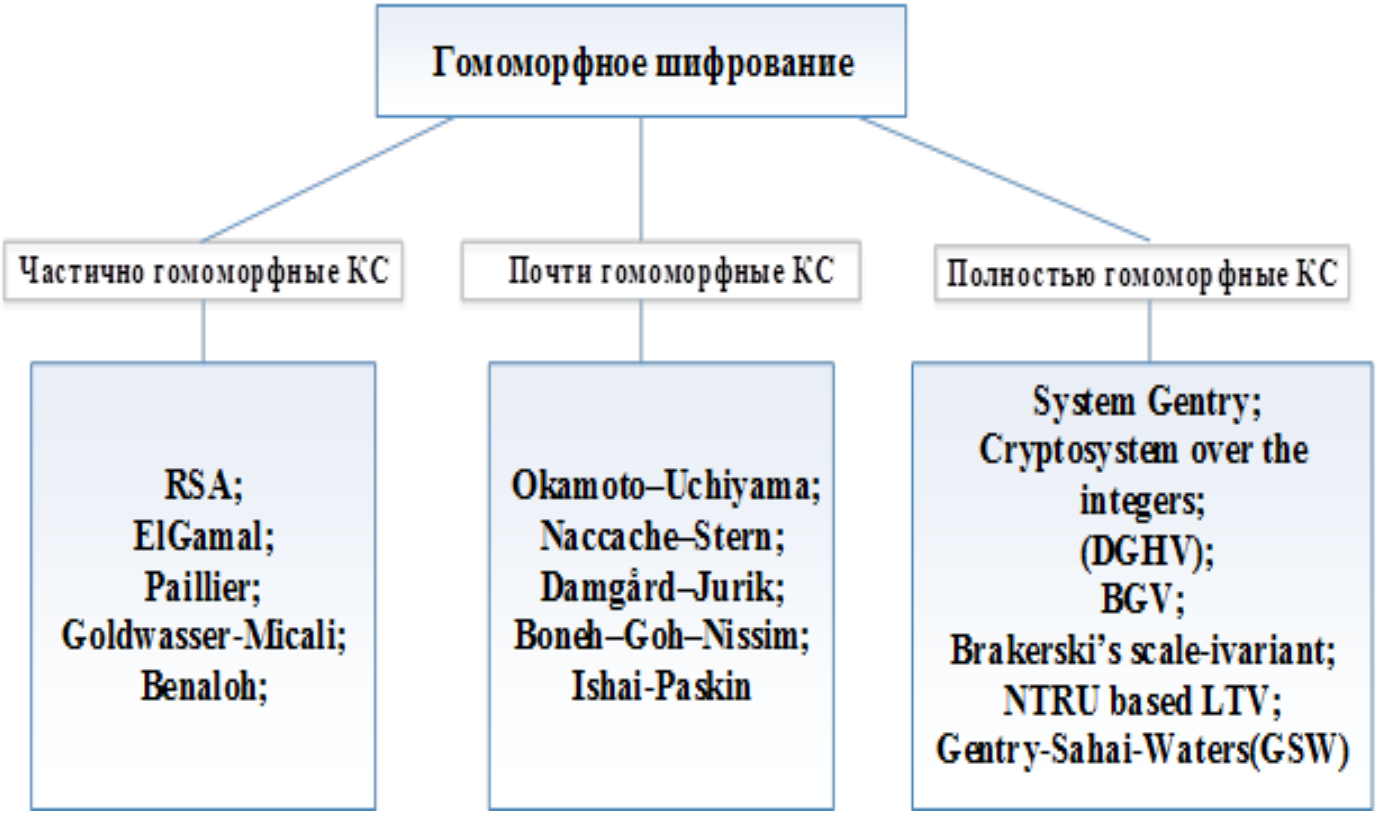
$$D(E(k,m_1)\ \xi 2\ E(k,m_2))= m_1\ \xi 1\ m_2$$

Гомоморфное шифрование может найти широкое применение в защите информации.

Пример. Вычисления над зашифрованными данными.

Пусть конфиденциальные данные хранятся в зашифрованном виде и над ними нужно произвести какие-то вычисления (сложить, умножить и пр.). Классический подход такой: данные нужно расшифровать, провести соответствующую операцию и снова зашифровать результат. Для этого нужна защищенная аппаратура, ключи и выполнение всех организационных мероприятий по защите данных. То есть все очень ответственно, строго и сложно. Избежать этого можно, если вычисления проводить над зашифрованными данными. А расшифровать, если нужно, получить конечный результат.

Классификация гомоморфных систем шифрования



Разрешено использование только одной операции в неограниченном количестве.

Можно использовать одновременно две операции, одна из которых будет выполняться в ограниченном количестве.

Применение обеих операций над зашифрованными данными в неограниченном количестве.

Примеры криптосистем для гомоморфного шифрования

1. Криптосистема РША

Обозначения N=pq –модуль;

e –открытая экспонента; (N,e) – открытый ключ,

d – закрытая экспонента.

C=E((N,e),m)=$m^e mod N$ –криптограмма.

D(d,E((N,e),m))=$C^d mod N$=m.

Функция E - **гомоморфна относительно операции умножения открытых текстов**. Действительно, для двух открытых сообщений m1, m2 и любого открытого ключа e криптограмма произведения равна произведению криптограмм сомножителей

D(d(E(e,m_1)·E(e,m_2))=m_1·m_2.

Покажем гомоморфизм по умножению для любых двух m_1 и m_2:

Пусть $c_1 = m_1^e \bmod N$, $c_2 = m_2^e \bmod N$.

$c_1 c_2 = m_1^e \bmod N \cdot m_2^e \bmod N = m_1^e \cdot m_2^e \bmod N = (m_1 \cdot m_2)^e \bmod N = c$.

D(c)= $((m_1 \cdot m_2)^e)^d) \bmod N = (m_1 \cdot m_2)$.

Пример,

Предположим, что имеются сообщения $m_1 = 56947$, $m_2 = 64413$.

Ключи: $pk = e = (5437, 189781)$, $sk = (49269)$.

Зашифруем сообщения, используя открытый ключ:
$c_1 = 56947^{5437} \bmod 189781 = 96068$; $c_2 = 64413^{5437} \bmod 189781 = 149380$.

$c_1 \cdot c_2 = 96068 \cdot 149380 \bmod 189781 = 157744$.

$D(c_1 \cdot c_2) = 157744^{49269} \bmod 189781 = 39943$.

Такой же результат мы можем получить при умножении открытых сообщений: $m1 \cdot m2 = 56947 \cdot 64413 \bmod 189781 = 39943$.

2. Криптосистема Эль -Гамаля

Генерация ключей

- Для генерации ключей проводятся следующие операции:
- Генерируется просто число p и примитивный элемент $g \in GF(p)$;
- Выбирается случайное число a, такое что

$1 \leq a \leq p-2$ и вычисляется значение g^a;

Открытый ключ - набор $y=(p, g, g^a mod\ p)$,

Закрытый ключ – число a.

Шифрование

1. Сообщение M представляется в виде цепочки чисел M_i, каждое из которых не превосходит $p-1$;
2. Выбирается случайное число k, $1 \le k \le p-2$;
3. Вычисляется $\gamma = g^k mod p,\ \delta = M_i (g^a)^k mod p$;
4. Получена криптограмма $E = (\gamma, \delta)$.

Дешифрование

1. С использованием закрытого ключа, вычисляется $\gamma^{-a} mod p$;
2. Восстанавливается сообщение $M_i = (\gamma^{-a}\delta\ ) mod p$.

$$\gamma^{-a}\delta = g^{-ak}\, M_i g^{ak} = M_i \, mod\ p.$$

Гомоморфное свойство системы Эль-Гамаля

Предположим, что мы имеем два зашифрованных сообщения C_1 и C_2 алгоритмом ElGamal,

$$C_1 = (\gamma_1, \delta_1) \qquad C_2 = (\gamma_2, \delta_2)$$

Найдем произведение криптограмм:

$$C_1 C_2 = (\gamma_1, \delta_1)(\gamma_2, \delta_2) = (\gamma_1 \gamma_2 . \delta_1 \delta_2)$$

Расшифруем общую криптограмму:

$$(\gamma_1 \gamma_2)^{-a} \delta_1 \delta_2 = \gamma_1^{-a} \delta_1 \cdot \gamma_2^{-a} \delta_2 = m_1 \cdot m_2$$

КС Эль гамаля гомоморфна относительно операции **перемножения** открытых сообщений

КС Эль-Гамаля с аддитивным гомоморфизмом

Шифрование

1.Сообщение M представляется в виде цепочки чисел M_i, каждое из которых не превосходит $p-1$;

2.Выбирается элемент поля b, имеющий порядок > |M|

3.Выбирается случайное число k, $1 \le k \le p-2$;

4.Вычисляется $\gamma = g^k mod p$, $\delta = b^{M_i}(g^a)^k mod p$;

5.Получена криптограмма $E = (\gamma, \delta)$.

Дешифрование

1. С использованием закрытого ключа, вычисляется $\gamma^{-a} mod p$;

2.Восстанавливается сначала $b^{M_i} = (\gamma^{-a}\delta)mod p$.

$$\gamma^{-a}\delta = g^{-ak}\, b^{M_i} g^{ak} = b^{M_i} mod\ p.$$

3. Затем путем логарифмирования восстанавливается сообщение M_i

$$M_i = \log_b (b^{M_i})$$

Гомоморфное свойство системы Эль-Гамаля с аддитивным гомоморфизмом

Предположим, что мы имеем два зашифрованных сообщения C_1 и C_2 алгоритмом Эль-Гамаля,

$$C_1 = (\gamma_1, \delta_1) \qquad C_2 = (\gamma_2, \delta_2)$$

Найдем произведение криптограмм:

$$C_1 C_2 = (\gamma_1, \delta_1)(\gamma_2, \delta_2) = (\gamma_1\gamma_2 . \delta_1\delta_2) = (\gamma_1\gamma_2, b^{M_1} g^{ak} \cdot b^{M_2} g^{ak})$$

Расшифруем общую криптограмму:

$$(\gamma_1\gamma_2)^{-a}\delta_1\delta_2 = \gamma_1^{-a}\delta_1 \cdot \gamma_2^{-a}\delta_2 = g^{-ak} b^{M_1} g^{ak} \cdot g^{-ak} b^{M_2} g^{ak}) = b^{M_1} b^{M_2}$$

$$\log_b (b^{M1} b^{M2}) = M_1 + M_2$$

Эта КС Эль-Гамаля гомоморфна относительно операции сложения открытых сообщений

Системы Рабина и ее гомоморфизм

- Закрытый ключ простые числа p,q
- Открытый ключ n=pq

Шифрование $C=M^2 \mathrm{mod} n$

Дешифрование $r = \sqrt{C} \bmod p \quad s = \sqrt{C} \bmod q$

и далее решение 4-х систем уравнений, получение 4-х решений: M1, M2, M3, M4.

- **Гомоморфное свойство КС Рабина:**

Предположим, что мы имеем два зашифро-ванных сообщения C_1 и C_2 алгоритмом Рабина,

$$C_1 = (m^2{}_1 \bmod n) \qquad C_1 = (m^2{}_2 \bmod n)$$

Найдем произведение криптограмм:

$$C_1 C_2 = (m_1^2 \bmod n)(m_2^2 \bmod n) = (m_1 m_2)^2 \bmod n = M1^2 \bmod n$$

Расшифруем общую криптограмму:

$$(C_1 C_2)^{1/2} \bmod n = (M1, M2, M3, M4)\,.$$

Одно из решений является истинным *Mi=m1m2*

3.Криптосистема Пэйе (Paillier 1999г.)

Математические основы

- НОД(a,b)-наибольший общий делитель - gcd(a,b)
- НОК(a,b) – наименьшее общее кратное (lcm(a,b)) – наименьшее целое, которое делится на оба числа без остатка

  lcm(4,6)=12, lcm(4,14)=28.

Замечательный факт lcm(a,b)=ab/gcd(a,b).

Пусть n=pq, p и q - простые числа . Тогда функция Кармайкла

$$\lambda(n) = lcm(p-1, q-1)$$

Z_n - множество целых чисел по модулю n,

Z_n^* -множество целых чисел взаимно простых с n, это множество состоит из чисел $\varphi(n)$,

$Z_{n^2}^*$ - множество целых чисел взаимнопростых с n^2 -это множество состоит из $n\varphi(n)$ чисел.

Пример построения подмножеств

Z_n : 0, 1, 2, 3, 4, 5 n=6

Z_n^* : 1, 5 $\varphi(n) = 2$

$Z_{n^2}^*$: 1, 5, 7, 11, 13, 17,19, 23, 29, 31, 33, 35 $n^2 = 36$

$$n\varphi(n) = 6 \cdot 2 = 12$$

3.Криптосистема Пэйе (Paillier 1999г.)

Генерация ключей

1. Выбираются p и q — два больших простых числа, такие что $\gcd(pq, (q -$

$$\lambda = \frac{(p-1)(q-1)}{\gcd(p-1, q-1)}$$

Способы выбора g

- 1. Случайно выбрать g из множества $Z^*_{n^2}$, удовлетворяющее условию

$$\gcd\left(\frac{g^{\lambda} \bmod n^2 - 1}{n}, n\right) = 1 \qquad (1)$$

Вероятность выполнения условия (1) высока.

- 2. Случайно выбрать α и β из множества Z^*_n, затем вычислить $g = (\alpha n + 1)\beta^n \bmod n^2$

В этом случае выбранное g всегда удовлетворяет условию (1).

Шифрование

- Для шифрования открытого текста m∈Zn выполняются следующие действия:

1. Выбирается случайное число r из Z_n^* ;
2. Вычисляется криптограмма по формуле:

$$Pai(m) = g^{\mathrm{m}} \cdot r^n mod n^2$$

Дешифрование

- Дешифрование криптограммы производится по формуле:
- $m' = L\left(Pai(m)^{\lambda} mod\ n^2\right)\mu\ mod\ n$.

Свойства гомоморфности

Криптосистема Пэйе обладает следующими свойствами гомоморфности:

1. при дешифровании произведения двух шифротекстов будет получена сумма соответствующих им открытым текстам:

- $D(Pai(m_1) \cdot Pai(m_2) mod\ n^2) = (m_1 + m_2) mod\ n$;

Частный случай: $D(Pai(m_1) \cdot g^{m_2} mod\ n^2) = (m_1 + m_2) mod\ n$;

2. при дешифровании криптограммы, возведенной в степень $d\ \epsilon\ Z_n^*$, будет получено произведение открытого текста и показателя степени d:

- $D(Pai(m))^d) mod\ n^2 = d \cdot m\ mod n.$

Частный случай: $D(Pai(m1))^{m2}) mod\ n^2 = m2 \cdot m1\ mod n.$

Пример схемы Пэйе

1. Генерация ключей

Выберем два больших различных простых числа $p = 7$ и $q = 5$ и проверяем условие $gcd\big(pq, (p-1)(q-1)\big) = 1.$

Вычисляем $n = pq = 7 \times 5 = 35$, $n^2 = 1225$ и $\lambda = lcm(6,4) = 12.$

Выбираем случайное целое число g, такое что $g \in Z^*_{n^2}$, $g = 3$.

Находим $\mu = \Big(L\big(g^{\lambda} mod\ n^2\big)\Big)^{-1}\ mod\ n = 29.$

Таким образом. найдены:

$(g, n) = (3,35) -$ открытый ключ,

$(\lambda, \mu) = (12,29) -$ закрытый ключ.

2. Шифрование m=8

- Выбираем произвольное $r \in Z_n^*, r = 9,$
- Вычисляем

$c = g^m \times r^n \, mod \, n^2 = 3^8 \times 9^{35} \, mod \, 1225 = 436 \times 949 \, mod \, 1225 = 939.$

- 3. Расшифрование
- Получена криптограмма c=939 c ∈ Z_{1225}

- Вычисляем $m' = L\left(c^{\lambda}\ mod\ n^2\right) \times \mu mod\ n = L(939^{12}\ mod\ 1225) \times 29\ mod\ 35 = 22 \times 29\ mod\ 35 = 8.$

- m'=m

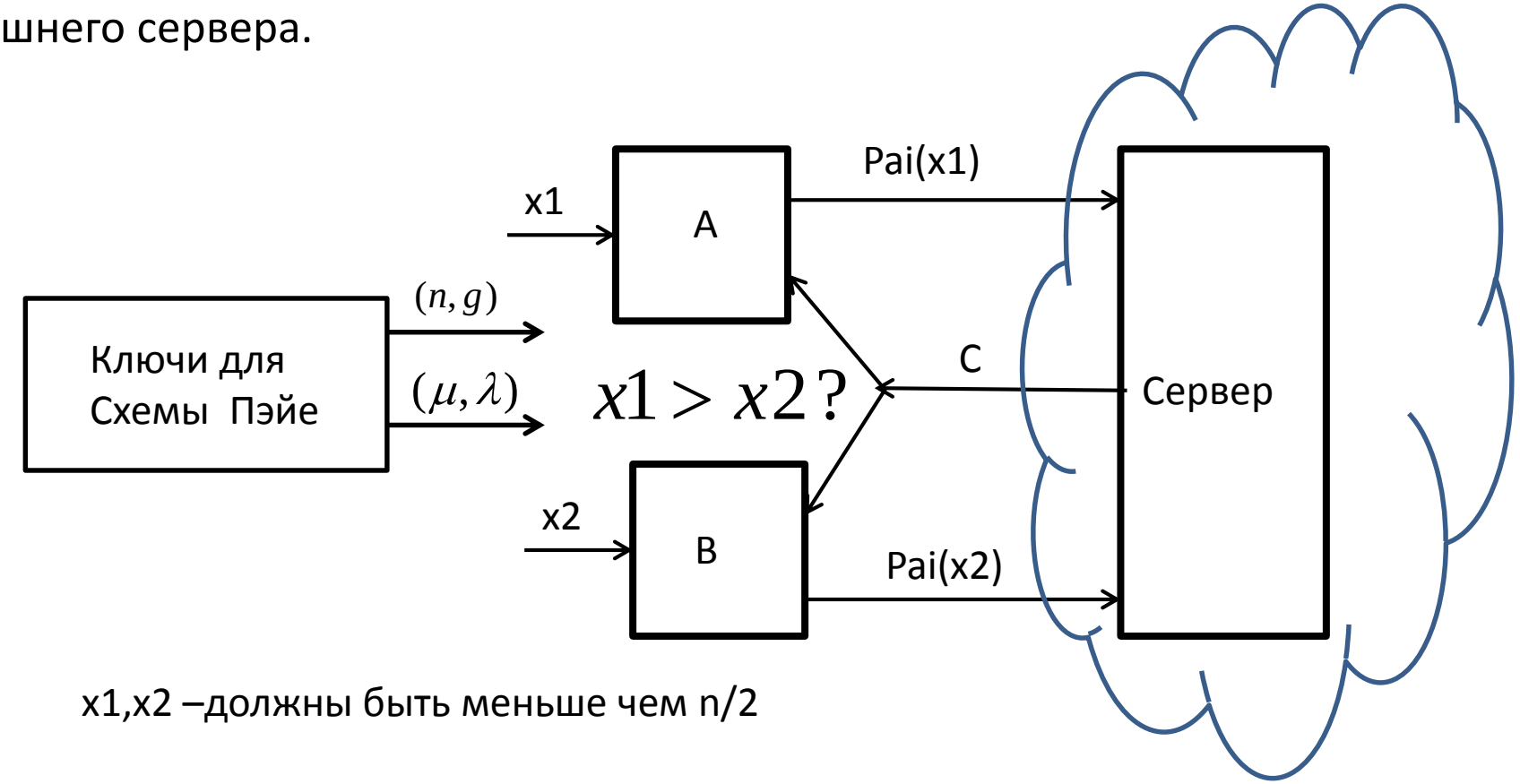
Применение схемы Пэйе

1. Анонимные вычисления.
2. Скрытый поиск точек интереса пользователей (point of interest –POI) пользователей.
3. Скрытое определение взаимных расстояний.
4. Тайное голосование.

Анонимные вычисления

Постановка задачи. Два пользователя A и B имеют числа x1 и x2 соответственно) и хотят выяснить у кого число больше, не раскрывая самих значений этих чисел.

Общая идея решения – использование гомоморфного шифрования и внешнего сервера.



x1,x2 –должны быть меньше чем n/2

Решение.

1. Пользователи получают ключи по схеме Пэйе: (n, g) - открытый ключ и (μ, λ) - закрытый ключ и случайное число – k.

2. Пользователь А шифрует число x1 по схеме Пэйе:

$$Pai(x1) = g^{x1} \cdot k^n \bmod n^2$$

3. Пользователь В шифрует число x2 по схеме Пэйе:

$$Pai(x2) = g^{x2} \cdot k^n \bmod n^2$$

4. Сервер выполняет преобразование зашифрованных данных

$$C = Pai(x1)Pai(x2)^{n-1} g^r$$

где r (r>0) - случайное число и отправляет С пользователям.

5. Пользователи А и В дешифруют С и по свойству гомоморфности получают:

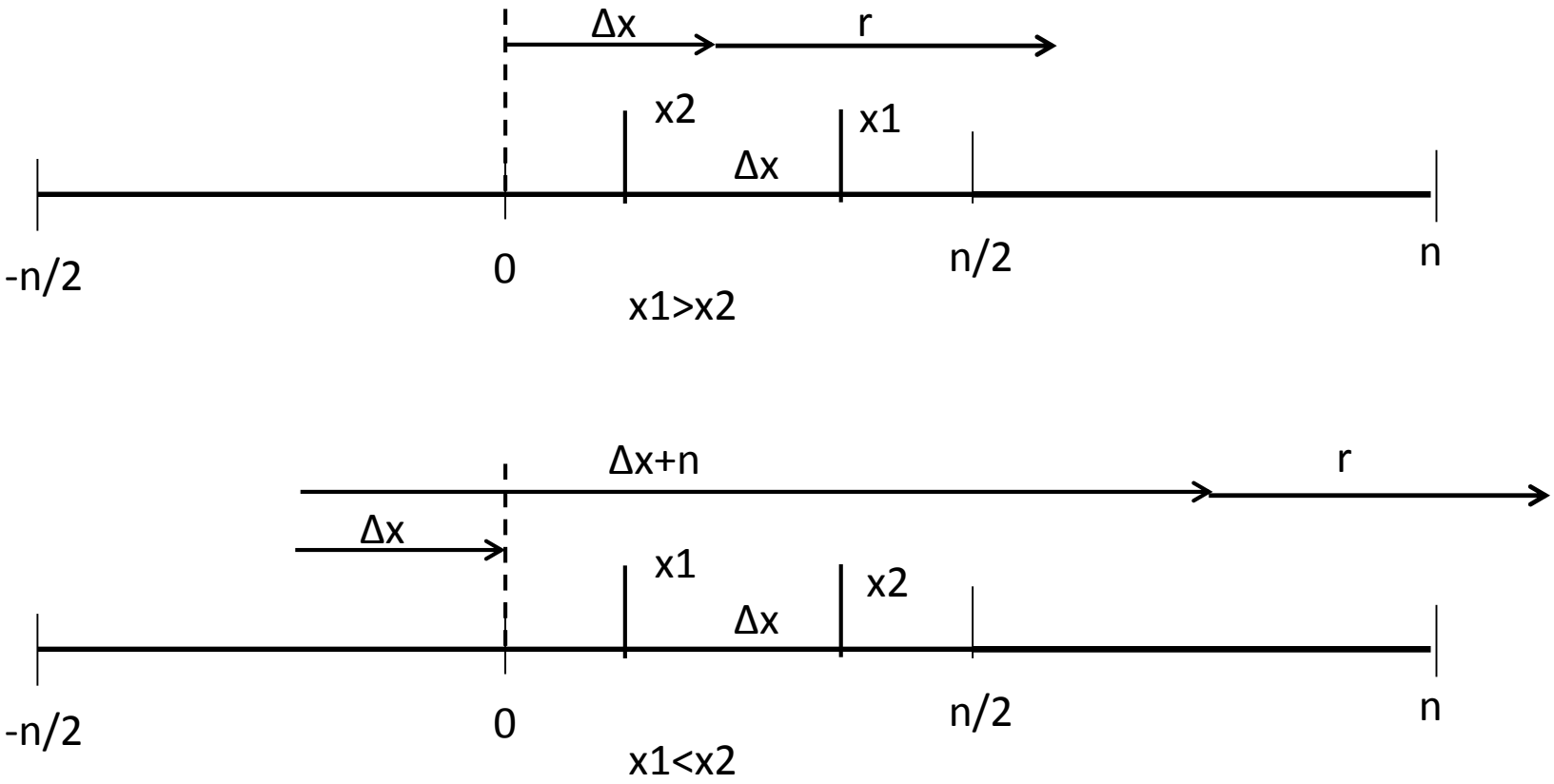
$$D(C) = Z = L(C^{\lambda} \bmod n^2)\mu \bmod n$$

По свойству гомоморфности

$$Z = (x1 + (n-1)x2 + r) \bmod n = (x1 - x2 + r) \bmod n = r + \Delta x$$

6. Тогда, если $Z > n/2, \quad mo\ x1 > x2$

если $Z < n/2, \quad mo\ x1 < x2$



Дополнительное задание к ЛР7

1. x1=№+10, x2 брать в одном случае меньше x1, в другом случае больше x1.

2. $n/2 < r < n$.

Протокол скрытого поиска точек интереса

Понятие объекта “Точки интереса”

Точки интереса или POI (от англ. points of interest) - это объекты инфраструктуры, достопримечательности, природные объекты, координаты и важные точки на дорогах, информация о которых нанесена на GPS карту.

- Например, к POI относятся: гостиницы, рестораны, АЗС, больницы, магазины, кинотеатры, музеи, банкоматы, аптеки и множество других объектов. Также к точкам POI относятся станции метро, вокзалы, аэропорты и прочие транспортные узлы. Отдельно выделяются дорожные POI: это посты ДПС, "лежачие полицейские", камеры видеонаблюдения, радары, железнодорожные переезды и прочие зоны повышенного внимания. Точки POI могут сопровождаться аудио предупреждениями.

Сервис определения местоположения

Карта с точками интереса пользоватеоя

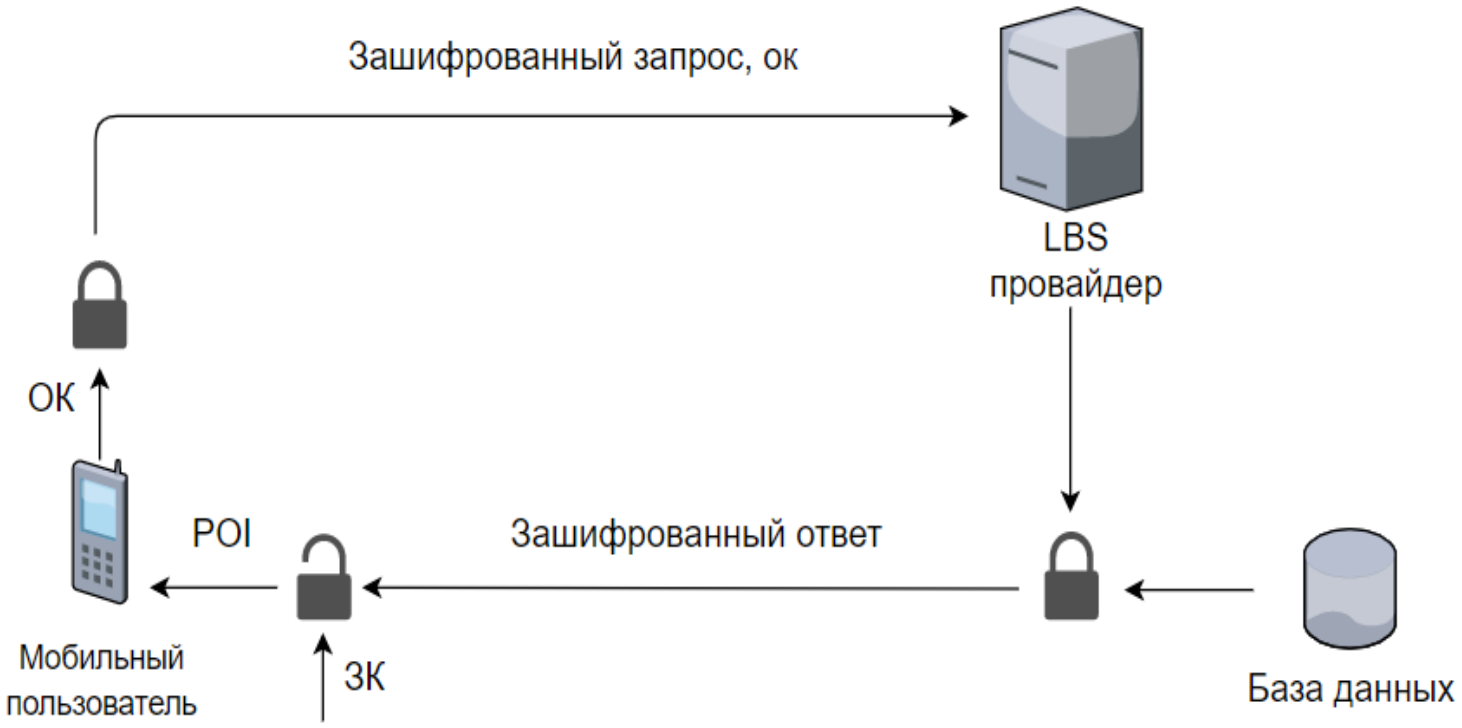
Взаимодействие пользователя и сервера LBS (Location-based service) – сервис определения местопололожения.

- LBS дает возможность пользователям запрашивать детальную информацию о точках интереса в их окрестности.

- Предположим, что мобильный пользователь хочет запросить у LBS провайдера информацию о k ближайших точках интереса. Для этого он должен предоставить LBS свое местоположение, на основании которого LBS вернет пользователю k ближайших POIs, сравнивая расстояния между пользователем и ближайшими точками интереса. Однако, при этом раскрывается местоположение мобильного пользователя LBS провайдеру.

- Данные о местоположении пользователей, даже когда пользователь не знает этого, позволяют получить намного больше информации, чем широта и долгота пользователя. Зная, где находится пользователь, можно узнать, чем он занимается, с кем, где и как часто проводит свое время. При получении этих данных можно определить привычки и повседневные дела пользователя и в какой момент он от них отклонился.

Протокол взаимодействия пользователя и LBS, обеспечивающий скрытность запрашиваемых точек интереса и координат пользователя (для краткости – протокол скрытого определения точек интереса)



Постановка задачи

- Мобильный пользователь знает координаты своего местоположения, а LBS знает координаты всех POIs.
- Мобильный пользователь инициирует запрос, который состоит из его местоположения и типа интересующей точки интереса, **шифрует** запрос, используя свой открытый ключ и отправляет его LBS провайдеру. После получения запроса, сервер выполняет вычисления над зашифрованными данными и данными местоположений POIs, хранящимися в его базе данных, при помощи открытого ключа, полученного от пользователя вместе с запросом, и отправляет полученный ответ обратно пользователю.
- Пользователь, получив зашифрованный ответ от сервера, **расшифровывает** его своим закрытым ключом и получает данные об интересующих его POIs.

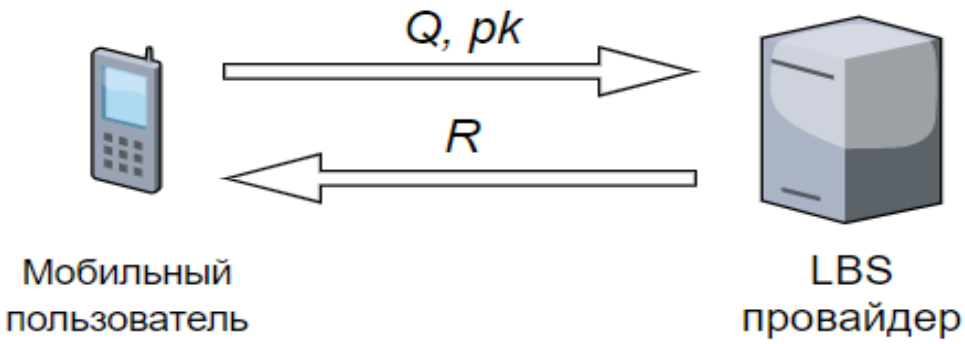
Типы протоколов скрытого определения местоположения “точек интереса” мобильных пользователей

1. Протокол скрытого определения местоположения точки интереса мобильного пользователя, не обеспечивающий конфиденциальность данных сервера;
2. Протокол скрытого определения местоположения точки интереса мобильного пользователя с конфиденциальностью данных сервера;
3. Протокол скрытого определения местоположения “точек интереса мобильного пользователя при сокрытии типа POI, обеспечивающий конфиденциальность данных сервера

Протокол скрытого определения местоположения точек интереса одного типа без обеспечения конфиденциальности данных сервера

(Протокол безопасного kNN-запроса)

- Позволяет абоненту найти ближайшие точки интереса в скрытой области без раскрытия своего местоположения.
- Построен с использованием гомоморфной схемы шифрования Пэйе

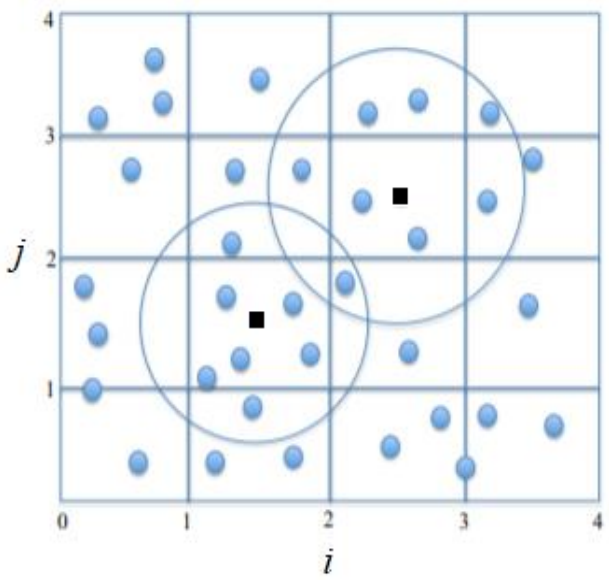


Протокол kNN-запроса состоит из четырех алгоритмов:

1. Генерация ключей;
2. Формирование запроса серверу;
3. Формирование ответа сервером;
4. Обработка ответа.

Подготовка исходных данных
(Протокол безопасного kNN-запроса)

- LBS провайдер разделяет базу данных местоположений пользователей (географическую карту) на ячейки одинакового размера, например, 1 км шириной и 1 км длиной. Базируясь на центре каждой ячейки, LBS провайдер определяет k ближайших точек интереса $P_1, P_1, \ldots, P_k$
- Ячейка разбивается на подячейки и каждая точка кодируется парой координат (x,y) , где и x – широта и y -долгота подячейки, в которой находится POI. Координаты POIs, представленные в виде последовательности бит, кодируются как целое число d_{ij} Предположим, что $M = max(d_{i,j})$ – самая длинная запись.
- Предположим, что мобильный пользователь U хочет найти k ближайших точек интереса вокруг своего местоположения. Пусть пользователь U находится в ячейке *(i,j)* .



Например запись *dij*=35_{10}=100011_2 обозначает координаты POI(x,y)=(x=100,y= 011) в ячейке *(I,j)*

Формирование запроса серверу

Пусть l – множество целых чисел от 1 до n. Для каждого $l \in \{1, 2, \dots, n\}$ пользователь выберает случайное целое число $r_l \in Z^*_N$ и вычисляет криптограммы c_l:

$$c_l = \begin{cases} Encrypt\ (1, pk) = g^1 r_l{}^N (mod\ N^2) \text{ если } l = i \\ Encrypt\ (0, pk) = g^0 r_l{}^N (mod\ N^2) \text{ если } l \neq i\ ' \end{cases}$$

где i – первая координата ячейки, в которой находится пользователь.

Таким образом, пользователь зашифровывает 1, если $l = i$, и 0 в любом другом случае, используя алгоритм шифрования криптосистемы Пэйе.

На выходе получаем: Зашифрованный запрос - $Q = \{c_1, c_2, \dots, c_n\}$, открытый ключ - pk. Зашифрованный запрос и прикрепленный к нему открытый ключ мобильный пользователь отправляет серверу.

Таким образом, запрос – это набор криптограмм, i-ая из которых содержит метку об i-ой координате пользователя.

Формирование ответа сервера

Сервер получает запрос $Q = \{c_1, c_2, \dots, c_n\}$ и открытый ключ $pk = \{g, N\}$.

D – база данных местоположения, которая содержит информацию о точках интереса и их расстоянии для каждой ячейки CR.

Q – зашифрованный запрос пользователя, посланный на сервер.

pk – открытый ключ, полученный от пользователя вместе с запросом.

На основе CR, n и D сервер вычисляет $R = \{C_1, C_2, \dots C_t, \dots, C_n\}$, где для $t = \{1,2, \dots, n\}$:

$$C_t = \prod_{l=1}^{n} {c_l}^{d_{l,t}} \ (mod\ N^2)\ ,$$

d11	d21	d31
d12	d22	d32
d13	d23	d33

то есть:

$$\mathrm{C}_1 = \ (\mathrm{c}_1{}^{d_{1,1}} \cdot \mathrm{c}_2{}^{d_{2,1}} \cdot \dots \cdot \mathrm{c}_n{}^{d_{n,1}})\ mod\ N^2;$$

$$\mathrm{C}_2 = \ (\mathrm{c}_1{}^{d_{1,2}} \cdot \mathrm{c}_2{}^{d_{2,2}} \cdot \dots \cdot \mathrm{c}_n{}^{d_{n,2}})\ mod\ N^2;$$

$$\vdots$$

$$\mathrm{C}_n = \ (\mathrm{c}_1{}^{d_{1,n}} \cdot \mathrm{c}_2{}^{d_{2,n}} \cdot \dots \cdot \mathrm{c}_n{}^{d_{n,n}})\ mod\ N^2.$$

где $d_{i,j}$ – координаты $(x\ , y\ )$ точек интереса пользователя относительно центра ячейки (i, j).

Таким образом, ответ состоит из n криптограмм, содержащих информацию о всех POI для каждой возможной координаты j пользователя.

Прием ответа R и его обработка

Из вектора R пользователь выбирает только Cj , где j – вторая координата ячейки, в которой находится пользователь. Все остальные данные, полученные от сервера, пользователь может игнорировать, так как
только Cj содержит информацию о k ближайших POIs для ячейки (i, j).
Пользователь вычисляет:
$d = Decrypt(C\text{j} , sk)$.
где алгоритм дешифрования является алгоритмом дешифрования криптосистемы Пэйе,

Так как криптосистема Пэйе является аддитивно гомоморфной, то результатом расшифрования произведения криптограмм является сумма их изначальных сообщений. Отсюда следует, что:

$$D(C_j) = d_{1,j} \cdot D(c_1) + d_{2,j} \cdot D(c_2) + \cdots d_{i,j} \cdot D(c_i) + \cdots + d_{n,j} \cdot D(c_n) =$$
$$= d_{1,j} \cdot 0 + d_{2,j} \cdot 0 + \cdots + \underline{d_{i,j} \cdot 1} + \cdots + d_{n,j} \cdot 0 .$$

Следовательно, так как все множители, кроме $D(c_i)$ равны 0, при расшифровании ответа сервера пользователь получает вектор $d_{i,j}$ параметров POI относительно своей ячейки (i, j).

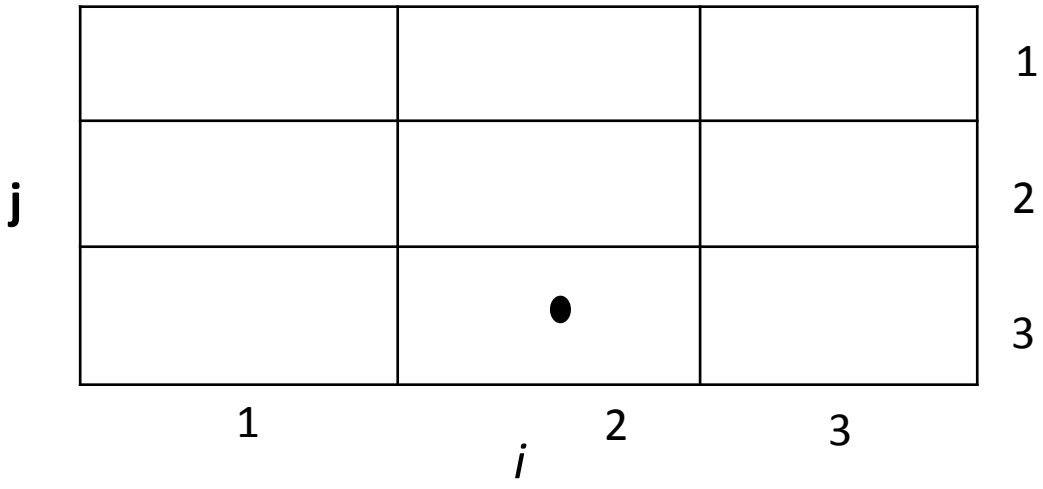
Протокол 1 (без конфиденциальности данных LBS провайдера), обеспечивает только скрытность местоположения мобильного пользователя, так как LBS провайдер не может определить его местоположение по полученному от пользователя зашифрованному запросу.

Пример протокола

Пусть скрытый регион разделен на 3 × 3 ячеек, пользователь находится в ячейке $(i, j) = (2, 3)$ и хочет получить информацию о k ближайших точках интереса относительно своей ячейки.

Пусть база данных местоположений ближайших точек интереса относительно центра каждой ячейки D представленна в виде матрицы чисел:

$$[dij] = \begin{bmatrix} 3 & 1 & 2 \\ 1 & 3 & 4 \\ 7 & 6 & 5 \end{bmatrix}.$$



Здесь i- абсциса,
j- ордината POI.
Не путать с обозначением индексов элементов матрицы!

Генерация ключей

Случайным образом выбираем два простых числа p и q:
Пусть p = 7, q = 11.
Вычисляем модуль N: $N = pq$ = 77.
Исходя из базы данных местоположений M = 7 поэтому такие значения p и q подходят.
Выбираем g из множества : $Z^*_{N^2}$:
Пусть g = 5774.
Открытый ключ $pk = \{g, N\}$ = {5774, 77},
Секретный ключ $sk = \{p, q\}$ = {7, 11}.

Формирование запроса (пример)

для каждого $l \in \{1, 2, \dots, n\}$ выбираем случайное число $r_l \in Z^*_N$.

Так как наша область имеет 3 × 3 ячеек, то $n = 3$.

Пусть $r1 = 12$, $r2 = 15$, $r3 = 17$.

Далее для каждого l вычисляем:

$$c_l = \begin{cases} Encrypt\,(1, pk) = g^1 r_l{}^N (mod\ N^2) \text{ если } l = i \\ Encrypt\,(0, pk) = g^0 r_l{}^N (mod\ N^2) \text{ если } l \neq i \end{cases}$$

$$c_1 = 12^{77} (mod\ 5929) = 3510;$$

$$c_2 = 5774 \cdot 15^{77} (mod\ 5929) = 776;$$

$$c_3 = 17^{77} (mod\ 5929) = 2175.$$

Получаем $Q = \{c_1, c_2, c_3\}, pk = \{g, N\}, s = sk = \{p, q\}.$

$$Q = \{3510, 776, 2175\}, pk = \{5774, 77\}\ sk = \{7, 11\}.$$

Генерация ответа сервером (пример)

На основе CR, n, D вычисляем $R = \{C_1, C_2, C_3\}$:

$$C_\gamma = \prod_{l=1}^{3} c_l^{d_{l,\gamma}} \ (mod\ N^2),$$

$$[d_{ij}] = \begin{bmatrix} 3 & 1 & 2 \\ 1 & 3 & 4 \\ 7 & 6 & 5 \end{bmatrix}$$

где $\gamma = 1, 2, 3$.

$C_1 = 3510^3 \cdot 776^1 \cdot 2175^2 (mod\ 5929) = 2847;$

$C_2 = 3510^1 \cdot 776^3 \cdot 2175^4 (mod\ 5929) = 1051;$

$C_3 = 3510^7 \cdot 776^6 \cdot 2175^5 (mod\ 5929) = 2613.$

Получаем сгенерированный ответ сервера $R = \{2847, 1051, 2613\}$.

Обработка ответа (пример)

Выбираем интересующую нас криптограмму C_3, так как $j = 3$

$$d_{2,3} = \frac{(C_3{}^{\lambda}(mod\ N^2) - 1)/N}{(g^{\lambda}(mod\ N^2) - 1)/N}(mod\ N),$$

где

$$\lambda = lcm(p-1, q-1) = \frac{(p-1)(q-1)}{\gcd(p-1, q-1)}.$$

$$\gcd(p-1, q-1) = 2.$$

$$\lambda = \frac{6 \cdot 10}{2} = 30.$$

Вычисляем d:

$$d_{2,3} = \frac{(2613^{30} mod\ 5929 - 1)/77}{(5774^{30} mod\ 5929 - 1)/77}\ mod\ 77 = 52 \cdot 60^{-1} mod\ 77 = 6$$

Полученный результат соответствует информации о точках интереса относительно ячейки (2,3).

Аналогично пользователь может расшифровать криптограммы C_1 и C_2.

Например, для $j = 2$ получаем:

$$d_{2,2} = \frac{(1051^{30} mod\ 5929 - 1)/77}{(5774^{30} mod\ 5929 - 1)/77}\ mod\ 77 = 26 \cdot 60^{-1} mod\ 77 = 3$$

Полученный результат соответствует информации о точках интереса ячейки (2,2).

Следовательно, конфиденциальность данных сервера этот протокол не обспечивает.

Для устранения этого недостатка разработан другой протокол, в котором пользователь дополнительно шифрует вторую координату j.

Таким образом, протокол на основе криптосхемы Пэйе отвечает требованиям безопасности:

- Только мобильный пользователь знает свое фактическое местоположение.
- Сервер выполняет вычисления только над зашифрованными данными и возвращает конечный результат мобильному пользователю в зашифрованном виде, тем самым сохраняя целостность и конфиденциальность информации.
- Злоумышленник не может иметь доступ к информации о местоположении пользователя, поскольку координаты отправляются на сервер в зашифрованном виде.
- Злоумышленник, пытающийся атаковать сервер, не может получить доступ к открытому тексту, поскольку у него нет секретного ключа.
- В применяемом алгоритме используется вероятностное шифровании открытого текста. При шифровании нового блока данных каждый раз генерируется новое случайное значение, что затрудняет дешифрование.