Советский суперкомпьютер К-340А и секретные вычисления

Кренделев Сергей Федорович
Новосибирский государственный университет

Работа выполнена при финансовой поддержке Минобрнауки РФ (договор № 02.G25.31.0054)

- Под секретными вычислениями подразумевается обработка зашифрованных данных без их дешифрования
- области применения таковы:
- Обработка персональных данных в облачных сервисах в зашифрованном виде
- Мощные методы обфускации на программном и аппаратном уровне.
- Защищенные базы данных в облаках, и на рабочем месте.

Считается, что для этих целей лучше всего подходит полностью гомоморфное шифрование.

Под гомоморфным шифрованием понимается любое отображение $\Phi:\Delta\to\Omega$, где Δ,Ω кольца произвольного вида, такое что $\Phi(x+y)=\Phi(x)+\Phi(y)$ $\Phi(x\times y)=\Phi(x)\times\Phi(y)$, если уравнение $\Phi(x)=\lambda$ имеет решение, то это решение единственно. Считается, что Φ - открытый ключ. Как решать уравнение $\Phi(x)=\lambda$ это секретный ключ.

- Литература на эту тему фактически является отчетами по грантам для DARPA (Defense Advanced Research Projects Agency — агентство передовых оборонных исследовательских проектов министерства обороны США),
- IARPA (Intelligence Advanced Research Projects Activity — Агентство по перспективным исследованиям разведывательного ведомства, национальная разведка США)
- Отчеты в компаниях IBM, Microsoft.

В этих работах рассматривается вариант гомоморфного шифрования, где $\Delta = Z_2$. Другими словами $\Phi: Z_2 \to \Omega$. Наличие гомоморфизма Φ позволяет секретно вычислять любые многочлены от любого числа переменных. Это означает, что можно имитировать (эмулировать) стандартный процессор с двоичным представлением данных и операциями, связанными с двоичной логикой. Это возможно, учитывая, что стандартные операции в поле Z_2 можно преобразовать в логические операции or, and (на самом деле достаточно nand).

Печальный факт заключается в том, что вариант с гомоморфным шифрованием $\Phi: Z_2 \to \Omega$ очень далек от практической реализации. Проблема в росте зашифрованных данных, особенно при умножении. В случае реализации логических схем умножение присутствует всегда

- Необходимы другие подходы, как к гомоморфному шифрованию, так и к компьютерным системам которые собираемся эмулировать.
- Существует вариант компьютеров основанных на системе остаточных классов (СОК) или в современных терминах на основе модулярной арифметике. Такие компьютеры разрабатывались в 50-70 годы прошлого столетия для создания РЛС. К ним относится суперкомпьютер К-340А, а также СуперЭВМ 5Э53, ЭВМ “Алмаз” и т.п. Все они основаны на модулярной арифметике. Особенностью данных компьютеров является очень высокий параллелизм, что позволяет делать вычисления в реальном времени.

- Недостатки – проверка выхода за диапазон данных, сравнение больше, меньше. С точки зрения разработчиков этих систем – “***модулярная и двоичная арифметика несовместимы”.*** Цель работы построить систему полностью гомоморфного шифрования, которая бы имитировала (эмулировала) этот класс вычислительных устройств.

Общие методы построения гомоморфного шифрования.

Стандартный метод построения гомоморфизмов колец проистекает из стандартной алгебраической геометрии. Пусть R произвольное кольцо, D произвольное множество. Рассмотрим множество функций определенных на множестве D со значениями в R, обозначим это множество $\Psi(D,R)$. Множество $\Psi(D,R)$ является кольцом относительно поточечного умножения. Теперь построим отображение из кольца R в кольцо $\Psi(D,R)$ по правилу, выбираем некоторую фиксированную точку $x_0 \in D$ (это секретный ключ), для всякого $\alpha \in R$ находим функцию $f \in D\Psi(D,R)$ такую, что $f(x_0) = \alpha$, функция f является представлением для элемента кольца α и объявляется как открытый ключ. Очевидно, что это гомоморфизм колец. Для определения зашифрованного элемента достаточно вычислить $f(x_0)$.

В практических приложениях обычно берут конечный набор функций $f_1, f_2, ..., f_r$, которые объявляются базисными, и рассматривают под кольцо в $\Psi(D,R)$ образованное всевозможными произведениями и суммами базисных функций.

Основная модель полностью гомоморфного шифрования.

Фактически вышеперечисленные компьютеры используют полностью гомоморфное шифрование. Согласно модулярной арифметике, если дан набор взаимно простых чисел $m_1, m_2, ..., m_k$ и $M = m_1 m_2 ... m_k$, то строится изоморфизм колец Z_M и $Z_{m_1} \times Z_{m_2} \times ... \times Z_{m_k}$. То, что это изоморфизм следует из китайской теоремы об остатках. Для ЭВМ К-340А набор модулей такой 2; 5; 23; 63; 17; 19; 29; 13; 31; 61. Первое, что приходит в голову, чтоб сделать из модулярного подхода гомоморфное шифрование это скрыть набор модулей $m_1, m_2, ..., m_k$, сделать из них секретный ключ. Однако такая система шифрования достаточно просто вскрывается атакой с известными данными.

Достаточно построить полностью гомоморфное шифрование для одного модуля, пусть этот модуль равен m. Тем самым рассматривается кольцо Z_m. Согласно алгебраической геометрии для построения гомоморфизмов необходимо построить множество функций со значением в кольце Z_m. Выберем простейший случай, в качестве множества функций выберем линейные функции n переменных

$$h(x_1, x_2, ..., x_n) = \alpha_1 x_1 + \alpha_2 x_2 + ... + \alpha_n x_n$$

$$\alpha_i \in Z_m \quad i=1,2,...,n$$

Множество таких функций обозначим R. К сожалению, R является модулем над кольцом Z_m, но не является кольцом. Для того чтобы сделать из множества R кольцо, введем набор n^3 структурных констант $\gamma_{ijk} \in Z_m \quad i,j,k=1,2,...,n$. Этот набор назовем таблицей умножения

Определим произведение двух функций

$$h_1(x_1, x_2, ..., x_n) = \alpha_1 x_1 + \alpha_2 x_2 + ... + \alpha_n x_n$$
$$h_2(x_1, x_2, ..., x_n) = \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n$$

По правилу

$$h_1(x_1, x_2, ..., x_n) \otimes h_2(x_1, x_2, ..., x_n) =$$
$$\sum_{k=1}^{n} x_k \sum_{i,j=1}^{n} \gamma_{ijk} \alpha_i \beta_j \quad .$$

Введение структурных констант позволяет снабдить множество R структурой алгебры (системы гиперкомплексных чисел). Умножение в этой алгебре не является, вообще говоря, коммутативным и ассоциативным.

Положим $\mathbf{u} = (u_1, u_2, ..., u_n) \in Z_m^n$, и определим гомоморфизм $\varphi : R \to Z_m$ по правилу

$\varphi(h) = h(u_1, u_2, ..., u_n)$, специализация (вычисление функции в точке).
$\mathbf{u} = (u_1, u_2, ..., u_n)$ назовем секретным ключом.

Утверждение 1. Структурные константы γ_{ijk} можно выбрать так, что отображение φ является гомоморфизмом колец.

Утверждение 2. Для любой секретной точки $\mathbf{u} = (u_1, u_2, ..., u_n)$ таблица умножения γ_{ijk} находится всегда, причем находится неоднозначно.

Таким образом, построено полностью гомоморфное шифрование в кольце Z_m. Секретным ключом является набор, состоящий из модуля m и точки $\mathbf{u} = (u_1, u_2, ..., u_n)$. Открытым ключом является таблица умножения $\gamma_{ijk} \in Z_m$ i,j,k=1,2,...,n.

Шифрование. Всякому $d \in Z_m$ сопоставляется функция

$$h(x_1, x_2, ..., x_n) = \alpha_1 x_1 + \alpha_2 x_2 + ... + \alpha_n x_n$$

$\alpha_i \in Z_m$ i=1,2,...,n

Такая, что $h(u_1, u_2, ..., u_n) = d \mod(m)$

Дешифрование. Всякой функции $h(x_1, x_2, ..., x_n)$ сопоставляется число

$$d = h(u_1, u_2, ..., u_n) \mod(m).$$

Гомоморфность данного шифрования вытекает из того факта, что множество линейных функций с заданной таблицей умножения не выводит из класса линейных функций.

В данном варианте не происходит увеличение размера данных при умножении

- Наличие таблицы умножения позволяет реализовать гомоморфное шифрование для рациональных чисел.
- Данная конструкция позволяет строить шифрование с открытым ключом из шифрования с секретным ключом. Этому приложению будет посвящен отдельный доклад.

СПАСИБО ЗА ВНИМАНИЕ.

Вопросы?