РусКрипто XXVIII НАУЧНО-ПРАКТИЧЕСКАЯ КОНФЕРЕНЦИЯ

ВЫСШАЯ ШКОЛА ЭКОНОМИКИ

Московский институт электроники и математики им. А.Н. Тихонова

Акционерное общество «Национальная система платежных карт»

Концепция конфиденциального обмена данными на основе модифицированных фильтров Блума для противодействия мошенническим схемам

Алмазбек уулу Тимур, студент программы ИБКФС МИЭМ ВШЭ
Шевченко Вячеслав Андреевич, студент программы ИБКФС МИЭМ ВШЭ
Сергеев Антон Валерьевич, доцент МИЭМ ВШЭ

Почему это важно?

Угрозы актуальны как никогда:

1) объем хищений у россиян в 2025 году составил 29,3 млрд. руб. по данным Банка России (+6,4% за год);
2) количество мошеннических операций составило 1,5 млн. (+31,2% за год);
3) опрос топ-10 банков говорит о том, что 94% всех атак производится с использованием техник социальной инженерии.

Мошенничество, как правило, направлено на несколько сервисов одновременно.

Невозможность взаимного обмена данными между различными организациями из-за:
1) регуляторных требований (152-ФЗ, ГОСТы и др.);
2) риска того, что централизованная база данных сама станет целью мошенников;
3) неготовности организаций делиться информацией с конкурентами.

Предлагается новый подход к борьбе с мошенничеством, учитывающий данные факторы

Как можно было бы решить проблему?

Централизованная база данных

Недостатки:

1. Сложность как сбора данных для такой базы, так и поддержание ее актуальности.
2. Централизованная БД – единая точка отказа.
3. Неопределенный статус оператора такой БД.
4. Медленная скорость проверки на принадлежность к БД

Хэширование

Хэширование данных о мошенниках и обмен хэшами между организациями.

Недостатки:

1. Потенциальная возможность перебора из-за небольшого универсума.
2. Низкая скорость поиска по хэш-таблице.

Классический фильтр Блума

Недостатки:

1. Отсутствие поддержки операции удаления.
2. Потенциальная уязвимость к атаке инвертирования.

Классический фильтр Блума

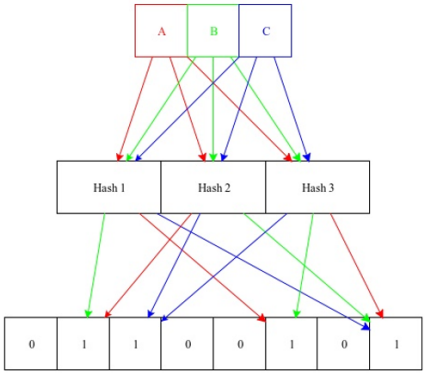
Фильтр Блума – вероятностная структура данных, позволяющая проверить принадлежность элемента множеству с определенной вероятностью ложноположительного срабатывания, но вероятностью ложноотрицательного срабатывания, равной нулю.

Основные характеристики:

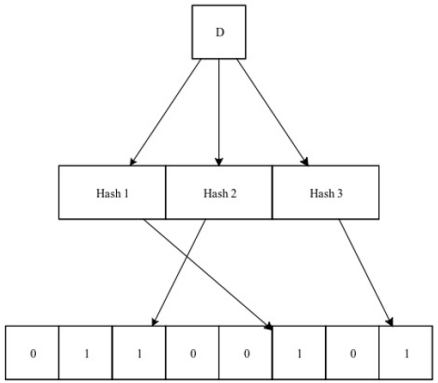
1) n – размер фильтра;
2) m – количество элементов, которое предполагается внести в фильтр;
3) $k = \frac{n}{m} \ln 2$ – количество хэш-функций;
4) $p = \left(1 - e^{-\frac{kn}{m}}\right)^k$ – вероятность ложноположительного срабатывания.

Преимущества:

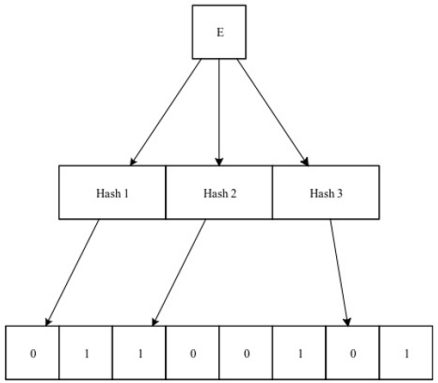
- не требует выделения большого количества памяти;
- поиск в фильтре выполняется за $O(k)$;
- несколько фильтров можно объединить посредством XOR;
- возможность контролировать точность;
- простота реализации.



Характеристика	Фильтр Блума	Хэш-таблица (идеальная)	Хэш-таблица (худший случай)	Бинарный поиск	Линейный поиск
Временная сложность	**O(k)**	O(1)	O(m)	O(log N)	O(N)
Пояснение	k - количество хэш-функций	Константа в среднем	m - число элементов (при множественных коллизиях)	N - количество элементов	N - количество элементов

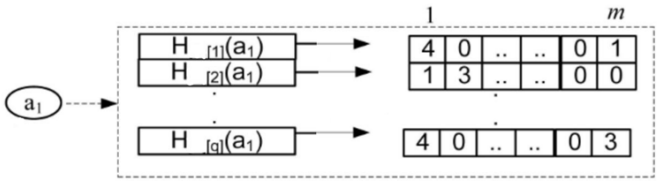


D возможно принадлежит множеству



E точно не принадлежит множеству

Варианты оптимизации фильтра Блума:



Тип фильтра	Ключевая оптимизация	Память	Удаление	Ложные срабатывания	Источник
Classic Bloom Filter	Базовый: битовый массив + k хэш-функций	~9.6-14 бит/элемент	Нет	~1%	[1, 7]
Counting Bloom Filter (CBF)	Счетчики вместо битов	~36-40 бит/элемент	Да	~1.3% (при 36 бит/элемент)	[2, 5, 6, 7]
Variable-Increment CBF (VI-CBF)	Переменные приращения вместо единичных	~36-40 бит/элемент	Да	<1.3% (при 36 бит/элемент)	[3, 7]
Scalable Bloom Filter (SBF)	Каскад фильтров; при заполнении создается новый фильтр большего размера с более низкой FPP	Возрастает динамически (например, ~60 МБ для 65,5 млн. эл-тов)	Нет	3%-30% (в зависимости от кол-ва элементов)	[4, 5, 7]
Parallel Bloom Filter	k независимых битовых массивов	~9.6-14 бит/элемент	Да	~1%	[4, 7]

Предлагаемая концепция:

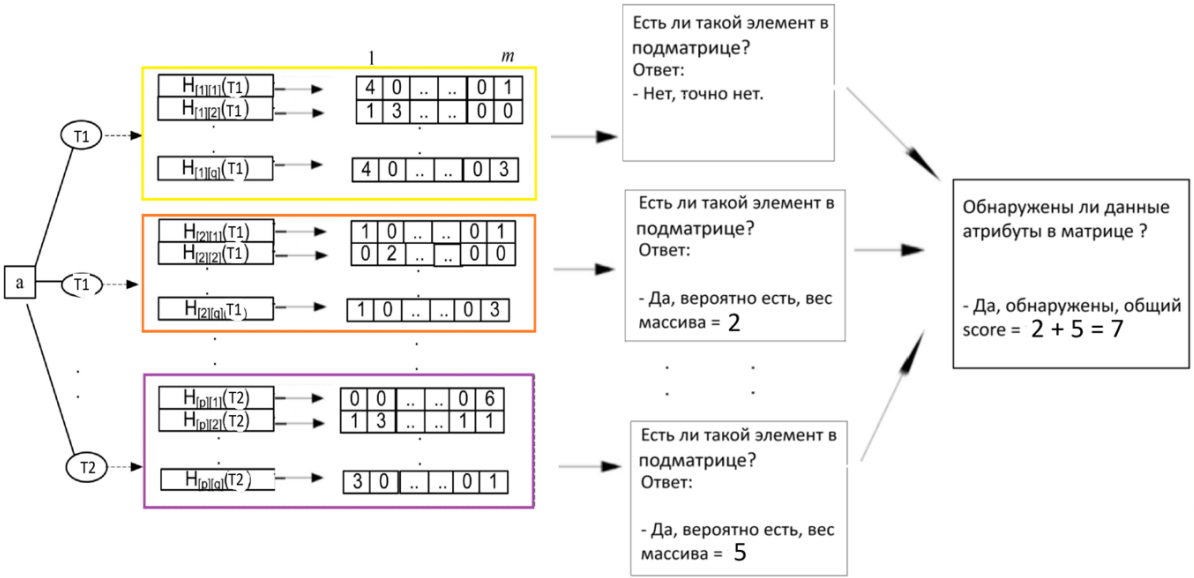
Пример:

Мошенник **a** имеет один номер телефона **T1,** привязанный к банковским приложениям, и другой номер телефона **T2,** привязанный к маркетплейсу.

Маркетплейс зафиксировал по номеру **T2** подозрительную активность - резкое появление большего количества объявлений и возвратов. Следовательно, как высокорисковый номер телефона, **T2** был добавлен в Фиолетовый подфильтр.

Приложение такси зафиксировало, что аккаунт, зарегистрированный на номер **T1,** часто ездит от одного банкомата к другому. Следовательно, как подозрительный, номер телефона **T1** был добавлен в Оранжевый подфильтр.

При попытке перевода денежных средств по номеру телефона с **T2** на номер телефона **T1** будет произведена проверка в нашем ФБ.



Например, пусть веса подфильтров будет:

- Желтый = 1
- Оранжевый = 2
- Фиолетовый = 5

Метод сокрытия данных путем хеширования (хэш цепочки)

Структура ячейки:

порядковый номер ячейки

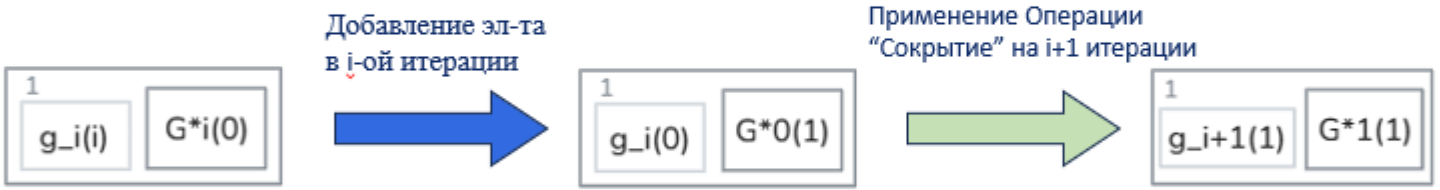
подячейка "счетчик",
где g_j(n) - хэш от n полученный j-ым методом хэширования
n - номер счета

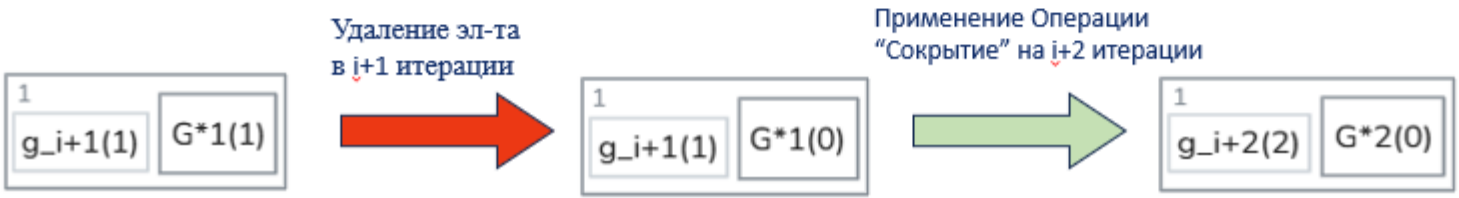
основная подячейка,
где G*n(x) = g_j(g_j-1(...(g_j-n(x)...))
- хэш цепочка от j-n до j-го метода хэширования
где j - номер текущей итерации

Операция «Сокрытие» - применение метода хэширования
- Добавление
- Удаление

подячейка "счетчик",
где g_j(n) - хэш от n полученный
j-ым методом хэширования
n - номер счета

основная подячейка,
где G*n(x) = g_j(g_j-1(...(g_j-n(x)...))
- хэш цепочка от j-n до j-го метода хэширования
где j - номер текущей итерации





Как осуществлять операцию проверки?

Для этого достаточно вычислить указывающие хэш-функции, тем самым узнав порядковые номера нужных ячеек, хранить хэши из основных подячеек с уже известными нам хэшами от нуля.

Почему важно только «не ноль» ?

Потому что если хотя бы одна из основных подячеек хранит 0, то элемента гарантированно нет в нашем ФБ

Разве не появляется уязвимость к частотному анализу?

(самый частый хэш в подячейке счетчика - хэш от наибольшего числа счета,
самый частый хэш в основной подячейки - хеш от 0, т.к. он наиболее часто встречается)

Операция “Саморефлексия” - функция, которая находит ячейки, содержащие нули, и ставит туда снова нули, но уже с другим номером счетчика (counter) = ∀ n, но не больше номера итерации. Соответственно, меняется номер счетчика и длина хэш-цепочки, хэширующей 0 в основных подячейках.

Повышается число итерации $\Rightarrow$ повышается длина хэш цепочек

Значительная длина хэш цепочек $\Rightarrow$ повышение трудоемкости

Предлагается ограничить число итерацией до 30 или 31 (число дней в месяце), после 31 будет исполняться **операция “Сброс итерации”,** которая сбрасывает порядок итерации.

Преимущества концепции обмена

1. Сохранение вероятностных характеристик относительно классического фильтра Блума.
2. Параллелизация операций.
3. Возможность распределенного хранения.
4. Поддержка динамического обновления.
5. Возможность настройки точности.

1. Конфиденциальный обмен информации.
2. Стойкость к переборным атакам.
3. Стойкость к атакам типа «человек посередине».
4. Защищенность базы к ее потере.
5. Возможность исключения участников.

Список литературы

1. Bloom B. H. Space/time trade-offs in hash coding with allowable errors // *Communications of the ACM*. 1970. Vol. 13, № 7. P. 422–426.
2. Fan, L., Cao, P., Almeida, J., & Broder, A. Z. (2000). Summary Cache: A Scalable Wide-Area Web Cache Sharing Protocol. IEEE/ACM Transactions on Networking, 8(3), 281–293. DOI: 10.1109/90.851975.
3. Rottenstreich, O., Kanizo, Y., & Keslassy, I. (2013). The Variable-Increment Counting Bloom Filter. *IEEE/ACM Transactions on Networking*, 22(6), 1850–1863. DOI: 10.1109/TNET.2013.2287124.
4. Almeida, P. S., Baquero, C., Preguiça, N., & Hutchison, D. (2007). Scalable Bloom Filters. *IEEE Global Telecommunications Conference (GLOBECOM 2007)*, 2758–2763. DOI: 10.1109/GLOCOM.2007.522.
5. Stoica & Pop (2025) — «Bloom Filters at Fifty: From Probabilistic Foundations to Modern Engineering and Applications».
6. Bonomi et al. (2006) — «An Improved Construction for Counting Bloom Filters».
7. Luo et al. (2018) — «Optimizing Bloom Filter: Challenges, Solutions, and Comparisons»

РусКрипто
XXVIII НАУЧНО-ПРАКТИЧЕСКАЯ КОНФЕРЕНЦИЯ

ВЫСШАЯ ШКОЛА ЭКОНОМИКИ

Московский институт электроники и математики им. А.Н. Тихонова

Акционерное общество «Национальная система платежных карт»

Спасибо за внимание!

Статья выполнена в результате проведения исследования в рамках Программы фундаментальных исследований Национального исследовательского университета «Высшая школа экономики» (НИУ ВШЭ).