Проверяемое разделение секрета и передача данных вслепую

Ю. Лифшиц*

30 ноября 2005 г.

План лекции

1. Проверяемое разделение секрета
2. Передача данных вслепую
3. Задача на дом

Начало

Введение

Сегодняшняя лекция является активной подготовкой к следующей лекции. В следующей лекции мы докажем самую крутую теорему в этом курсе. Это так называемая теорема о многосторонних секретных вычислениях, а сейчас мы будем решать мелкие задачи, которые будут, как бы являются подзадачами той большой задачи. Значит, у нас есть две задачки на эту лекцию одна называется "Проверяемое разделение секрета", другая называется "Передача данных в слепую". Ну для каждой задачи я объясню что нужно сделать, потом в первом случае мы сначала придумаем еще одну привязку к биту, не смотря на две, которые мы уже сочинили в лекции номер 1 и потом с помощью привязки к биту сделаем проверяемое разделение секрета. А во второй я расскажу два разных определения передачи данных вслепую. Сначала я расскажу, почему они эквивалентны, потом сделаю такой, предварительный протокол, в нем будет ошибка, и потом мы еч заделаем. И в конце лекции будет задача на дом. Задача на дом это открытый вопрос, который никто не знает как решать, которую предложил мне вчера Саша Куликов, вот я спешу поделиться...

*Законспектировал Е. Елизаров.

1 Проверяемое разделение секрета

1.1 Постановка задачи

У нас есть некоторое секретное число в некотором интервале целых чисел, скажем, от 1 до N, у нас есть n участников и мы хотим раздать некоторую информацию участникам так, что любые t из них могут восстановить секрет, но $t-1$ не могут сказать про него ничего. Если у нас три участника, то мы можем хотеть, что любые два из них могли восстановить секрет, но один ничего бы не узнал.

Формализация: Разделить секрет $m \in [1..N]$ между n участниками Любые t из них могут восстановить m Любые $t-1$ из них НИЧЕГО не могут узнать про m

Мы хотим сделать дополнительное требование: мы предполагаем такую ужасную вещь, что протокол, который мы зафиксировали для раздачи секрета вдруг стал кем-то нарушаться. Мы хотим сделать дополнительное требование, что если раздающий не по правилам раздает секрет, а присылает какие-то левые файлы, то тогда участники смогут это обнаружить. Причем воможно, что нечестен не только раздающий, но получающие.

1.2 Схема Шамира

Вспомним как действовала старая схема разделения секрета, схема Шамира. Значит, мы брали простое число p и считали, что все секреты это все возможные остатки по модулю p. Для того, чтобы закодировать секрет, мы брали случайный многочлен степени $t-1$, у которого случайным образом брали все коэффициенты, а свободный член был нашим секретом. $s_1, s_{t-1} \overset{\text{ran}}{\in} \mathbb{Z}_p$ $s(x) \overset{def}{=} m + s_1 x + \cdots + s_{t-1} x^{t-1}$ И как мы раздавали этот секрет? Мы раздавали значение многочлена в разных точках. (Леша больше не отвечает: у него хорошая память)!) Мы раздавали каждому участнику его номер (на всякий случай, чтоб он не забыл) и значение в этом номере. $(i, s(i))$ Нумеруем с единицы. Какое дополнительное требование мы хотим сделать к протоколу? Если бы у нас был честный раздающий, то он бы всем раздал значение в разных точках одного многочлена. Соответственно мы хотим это гарантировать. Каким бы не был раздающий, убедиться в том, что числа, которые он всем выдал, это точки на графике одного и того же многочлена с нужными абсциссами. Мы заставим раздающего доказать участникам, что он действительно сделал то, что он должен был сделать. Как мы это сделаем? Для того чтобы это сделать нужно сочинить новую привязку к биту. Сейчас мы ее будем сочинять. Вот такая неформальная задача, то есть у нас есть схема Шамира и теперь мы хотим сделать метод, который позволит участникам, пообщавшись между собой убедиться в том, что они все, без раскрытия друг другу своих данных, получили кусочки одного многочлена.

1.3 Еще одна привязка к биту

(Что такое привязка к биту скажет мне не Леша!) Что такое привязка к биту? Это было на первой лекции. Привязка к биту это задача, у нее есть участники, у них есть у каждого свои цели. Я не спрашиваю, как это делается, я спрашиваю, что делается и с какой целью. Я чувствую, вы хорошо работаете дома над лекциями, глубоко проникаете в их суть и заранее готовитесь к экзамену. Я думал, может откладывается что в память! (лол) Значит, напомню, у нас есть два участника Алиса и Боб. Они хотят промоделировать ситуацию в букмекерской конторе, у Алисы есть некоторая ставка, в простейшем случае команда выиграет или проиграет. Скажем волейбол нет даже ничьей. Она хочет так послать ставку Бобу, чтобы выполнялись два условия, которые назывались связность и секретность. Связность обозначает, что после того, как Алиса отдала ставку Бобу она не может ее изменить. А секретность, что Боб держит конверт запечатанным до конца соревнований, и открывает его только после окончания матчей, таким образом, доказывает Алисе, что не подкупил ни одну из команд. Это то, что было бы в букмекерской конторе. В нашем случае электронной привязки к биту (этот термин придумали москвичи) название странное, но я следую ему, так как это главный центр криптографии в России. У нас есть x это собственно выбор Алисы, проиграют или победят Она с помощью строчки случайных битов строчки r сосчитает функцию $C(x, r)$. Эта функция есть что-то типа шифрограммы ее выбора и посылает Бобу. Боб хранит ее, потом происходит соревнование, и после него - стадия открытия, по-английски это называется commitment (вот такая посылка). Открытие электронной ставки, когда Алиса раскрывает свой случайный выбор r и Боб может убедиться в том, что действительно $C(x, r)$, это то, что он получил и тогда он выплачивает выигрыш или наоборот не выплачивает.

Предъявляются два требования связанности и секретности. Некоторые обозначения для реализации данной схемы: $p = 2q + 1$ $p, q \in \mathbb{P}$ Таких чисел довольно много и их довольно легко сгенерировать. Число p будет иметь размер порядка 1000 бит, q порядка 500 бит. Мы будем рассматривать только остатки, которые являются квадратичными вычетами, то есть сравнимы с каким-нибудь квадратом. Скажем, по модулю 7 квадратичными вычетами будут $1, 4, 2$ потому,что $1^2 \equiv 1, 2^2 \equiv 4, 2 \equiv 3^2$. У нас есть простое число. Факт, на который мы будем опираться (это скорее не факт, а предположение) в нашей конструкции. То, что для данных квадратичных остатков g, h Трудно определить такое x, для которого $g^x \equiv h \mod p$. Все верят, что это трудная задача. Если p будет 1000 бит и будет два таких тысячебитных остатка, то неизвестно решение поиска степени, в которую надо возвести один, чтоб получить другой. Дискретное логарифмирование - это по h и g найти x. x называется дискретным логарифмом h по основанию g по модулю p. Значит, нам надо получить свойство гомоморфности.

Кто может сделать предположение, что такое гомоморфная привязка к биту? Алиса послала конверт с результатом на один матч, потом на второй матч. Боб сложил оба конверта и получил прогноз на сумму матчей. Боб

не может расшифровать каждый конверт по отдельности, но при этом может получить сумму прогнозов. (Если сумма, то привязка гомоморфна по сложению, если произведение, то по умножению.)

Итак, по шагам: Боб (букмекер) объявляет число $p = 2q + 1$ и остатки g и h, выбранные случайным образом. Дальше мы будем делать привязку к биту(Commitment) с помощью функции $C(x, r)$. $C(x, r) = g^x \cdot h^r$ Получился некий остаток по модулю p? его мы и послали Бобу. Вот собственно вся схема.

Теперь надо проверить два основных свойства: связность и секретность. Кто может какое-нибудь из этих свойств обосновать? Давайте начнем с секретности. Почему тот остаток, который получит Боб, не даст ему возможности узнать, чему был равен x? $\forall y \exists q \ : \ g^x \cdot h^r = g^y \cdot h^q$ Это утверждение верно, если g и h первообразные корни! h и g всегда первообразные корни: при $p = 2q + 1$ любой квадратичный вычет является первообразным корнем. Один и тот же конверт может содержать разные внутреннее сообщения. Связность: если Алиса зашифровала $C(x, r)$ и послала Бобу, то она не может выбрать другие y и q_1, так как-будто она зашифровала y. Если она может это сделать, то ей надо решить следующую задачу: $\log_g h \equiv (x - y)/(q_1 - r) \mod q$. Доказательство у доски :

$$g^{(x-y)} \equiv h^{(q1-r)}$$

$$\Updownarrow$$

$$x - y \equiv (q_1 - r) * log_g h \mod q$$

$$\Updownarrow$$

$$(x - y)/(q_1 - r) \equiv log_g h \mod q$$

По Th Эйлера любой остаток в степени $\varphi(p) = 2q$ дает единицу, поскольку мы рассматриваем квадратичные остатки их q-ая степень (Их корень квадратный в степени $2q$) обязательно единица. Т.е. по модулю 7 1, 2, и 4 в кубе уже дают единицу. У нас есть свойство гомоморфности, т.е., если у нас есть два конвертика, один с числом x другой с числом y. Если мы их перемножим, то мы перемножим $g^x \cdot h^r$ и $g^y \cdot h^q 1$ получится то же самое, что $g^{(x+y)} \cdot h(r + q_1)$. Это будет зашифрованным конвертиком, для числа $(x+y)$. Эту операцию Боб может сделать самостоятельно. $C(x, r) \cdot C(y, q) = C(x + y, r+q)$ это свойство гомоморфности. У нас есть новая привязка к биту, есть способ закодировать число x выбором случайного r и перемножением $g^x \cdot h^r$ Оно обладает абсолютной секретностью и вычислительной стойкостью.

Старая схема, тоже основанная на дискретном логарифме. Нам Боб присылал первообразный корень g, а мы по нему возвращали $C(b, r) = y^b \cdot g^r$. b было нулем или единицей. Схема очень похожая. Новая схема обладает гомоморфностью. А так же есть одно техническое отличие: эта схема коммитит только 0 и 1, а та коммитит остатки по модулю q. Гомоморфность относительно сложения выполняется тоже по $\mod q$. идея в следующем: Боб хочет послать Алисе такой остаток y, у которого Боб сам Знает дискретный логарифм, а Алиса - нет.

1.4 Контроль над раздающим

Теперь сделаем проверяемое разделение секрета. Раздающий подготавливает два многочлена: в первом секрет x и случайные коэффициенты, во втором все коэффициенты случайные. $f = x + f_1 z + \cdots + f_{t-1} z^{t-1}$ и $f' = f'_0 + \cdots + f'_{t-1} z^{t-1}$ Раздающий публикует привязки для $A_i \equiv g^{f_i} \cdot h^{f'_i} \mod p$. Например эти привязки вывешиваются на общедоступном сайте. На следующем шаге раздающий посылает каждому участнику три значения $(i, f_i), f'_i)$, где i - номер участника. Теперь каждый участник может выполнить проверку: $g^{f_i} \cdot h^{f'_i} \equiv A_0 \cdot A_1^i \cdot A_2^{i^2} \cdot \cdots \cdot A_{t-1}^{i^{t-1}} \mod p$ Доказательство на доске:

$$A(i) \equiv g^{f_i} \cdot h^{f'_i} \mod p$$

$$g^{f_i} \cdot h^{f'_i} \equiv (g^{f_0} \cdot h^{f'_0}) \cdot (g^{f_1 \cdot i} \cdot h^{f'_1 \cdot i}) \cdot \cdots \cdot (g^{f_{t-1} \cdot i^{t-1}} \cdot h^{f'_{t-1} \cdot i^{t-1}}) \mod p$$

Если проверка не прошла, участник сообщает всем остальным об этом. Участник считает раздающего жуликом, если: его собственная проверка не прошла, было объявлено о t нарушениях, менее t участников не объявляли о нарушениях, при этом число жуликов должно быть строго меньше половины числа участников.

2 Передача данных вслепую

2.1 Постановка Рабина

Мы рассмотрим две постановки данной задачи:

1. Постановка Рабина
2. Передача 1 из 2

У нас есть два участник: S Sender и R Resiver (отправитель и получатель соответственно). Постановка задачи состоит в том, что S посылает, бит b, а R с вероятностью $1/2$ получает b, и с вероятностью $1/2$ не получает ничего (он понимает, что не получил ничего). У этой задачи есть физический эквивалент: передача данных по шипящему каналу.

2.2 Передача 1-из-2

Авторы этого метода Even, Goldreich и Lempel. Lempel- человек, имя которого обозначает буква L в названии известного семейства алгоритмов сжатия данных.

В этой постановке так же участвуют S и R. Постановка задачи заключается в следующем: У S есть два бита b_0 и b_1, R надо получить i-тый бит (i - бит запроса). В результате R должен получить b_i, но не узнать ничего о b_{1-i},а S, в свою очередь, не должен узнать i.

Попробуем сделать из первой схемы вторую, и наоборот.

2.3 Схема Рабина $\Longrightarrow$ Схема 1-из-2

Итак, S генерирует случайным образом строчку из $3n$ битов $(a_1, \ldots, a_{3n})$ и отправляет их R по схеме Рабина (По "Шумящему"каналу). По постановке схемы Рабина до R дойдет примерно половина битов. R выбирает две непересекающихся группы индексов (причем биты с номерами одной из них он должен знать) $i_1, \ldots, i_n$ и $j_1, \ldots, j_n$ и посылает их S. S посылает R биты c_0 и c_1 такие, что $c_0 = a_{i_1} \oplus \cdots \oplus a_{i_n} \oplus b_0$, $c_0 = a_{j_1} \oplus \cdots \oplus a_{j_n} \oplus b_1$. Вероятность восстановить одновременно биты b_0 и b_1 экспоненциально мала (по схеме Рабина) И при этом S не может определить бит i.

2.4 Схема Рабина $\Longleftarrow$ Схема 1-из-2

Теперь, наоборот: у нас есть канал 1-из-2, надо организовать схему Рабина. Пусть S надо передать бит b. Надо сделать так чтобы R получил либо b, либо ничего, а S не узнал бы ничего об успешности передачи. Итак, S выбирает случайно бит a, далее с вероятностью $1/2$ запускает схему 1-из-2 для пары (a, b) и с вероятностью $1/2$ - схему 1-из-2 для пары (b, a), после чего отправляет R пояснение: какую схему он использовал. В результате R получает один бит. Причем с вероятностью $1/2$ это будет бит b, а с вероятностью $1/2$ - бит a. Причем S не знает, какой бит получил R т.к. не знает, какой бит из пары выбрал R (по схеме 1-из-2), и R понял что именно он получил (бит или ничего).

2.5 Протокол для 1-из-2

Создадим протокол для передачи данных по схеме 1-из-2. Будем использовать односторонние перестановки с секретом. S сообщает одностороннюю функцию f R. Тот ,в свою очередь, выбирает случайно x_i, считает $y_i = f(x_i)$ и выбирает случайное y_{1-i},после чего посылает получившиеся значения y_0 и y_1 S. S высылает $b_0 \oplus HCB(f^{-1}(y_0))$ и $b_1 \oplus HCB(f^{-1}(y_1))$. В результате R может восстановить только один бит, если он действительно случайно выбирал y_{1-i}. Попробуем заставить его выбирать действительно случайный бит.

Воспользуемся для этого ранее изученным нулевым разглашением. Заставим R доказать нам случайность битов. R выбирает случайно r_1, посылает $Q = C(r_1, t)$ для S, который отсылает другую случайную строку r_2. R использует $r_1 \oplus r_2$ в качестве y_{1-i} R доказывает с нулевым разглашением: $\exists t \exists r_1 \exists i \, : \, Q = C(r_1, t) \;\&\; y_i = r_1 \oplus r_2$

Таким образом, мы получили схему 1-из-2. Как мы доказали ранее из нее можно сделать схему Рабина.

3 Задача

Открытый вопрос от А. Куликова

Пусть есть граф из n вершин, степень каждой вершины не больше трех. Для какой наименьшей функции $f(n)$ всегда можно разбить вершины на две группы по $n/2$ так, чтобы между ними было не более $f(n)$ ребер?

Гипотеза: $f(n) = c \cdot n$ для некоторого c

Нижние оценки. Можете ли придумать граф, в котором в любом разрезе будет хотя бы $\log n$ ребер? (Задача имеет приложения в разработке эффективных алгоритмов.)

Итоги

Если не запомните ничего другого:

1. Проверяемое разделение секрета основано на гомоморфной привязке к биту.
2. Два подхода к передаче данных вслепую: Модель Рабина и 1-из-2.
3. Использовали нулевое разглашение для передачи данных вслепую.

Литература

Список литературы использованной при подготовке к лекции

1.