Проверяемое разделение секрета и передача данных вслепую

Лекция N 7 курса “Современные задачи криптографии”

Юрий Лифшиц
yura@logic.pdmi.ras.ru

СПбГУ — SPRINT Lab

Осень’2005

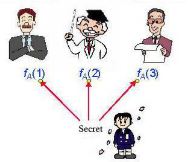
План лекции

1. **Проверяемое разделение секрета**
   - Постановка задачи
   - Еще одна привязка к биту
   - Протокол проверяемого разделения секрета
2. **Передача данных вслепую**
   - Две постановки
   - Базовый протокол
   - Форсируем случайность битов
3. **Задача**

План лекции

1. **Проверяемое разделение секрета**
   - Постановка задачи
   - Еще одна привязка к биту
   - Протокол проверяемого разделения секрета
2. Передача данных вслепую
   - Две постановки
   - Базовый протокол
   - Форсируем случайность битов
3. Задача

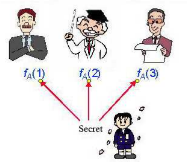
Проверяемое разделение секрета



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

Проверяемое разделение секрета



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

Дополнительное требование:

если раздающий нарушает протокол, честные участники смогут это обнаружить

Схема Шамира

Подготовительный шаг

Раздающий выбирает простое p, которое больше всех возможных секретов

Схема Шамира

Подготовительный шаг

Раздающий выбирает простое p, которое больше всех возможных секретов

Кодирование секрета

Выбираем $s_1, s_{t-1} \overset{\text{ran}}{\in} \mathbb{Z}_p$

Устанавливаем $s(x) \overset{def}{=} m + s_1 x + \cdots + s_{t-1} x^{t-1}$

Схема Шамира

Подготовительный шаг

Раздающий выбирает простое p, которое больше всех возможных секретов

Кодирование секрета

Выбираем $s_1, s_{t-1} \overset{\mathrm{ran}}{\in} \mathbb{Z}_p$

Устанавливаем $s(x) \overset{def}{=} m + s_1 x + \cdots + s_{t-1} x^{t-1}$

Раздача секрета

Для каждого $i = 1, 2, \ldots, n$

посылаем участнику i пару чисел $(i, s(i))$

Схема Шамира

Подготовительный шаг

Раздающий выбирает простое p, которое больше всех возможных секретов

Кодирование секрета

Выбираем $s_1, s_{t-1} \overset{\text{ran}}{\in} \mathbb{Z}_p$

Устанавливаем $s(x) \overset{def}{=} m + s_1 x + \cdots + s_{t-1} x^{t-1}$

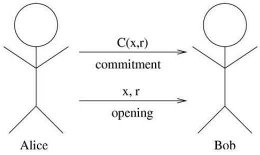
Раздача секрета

Для каждого $i = 1, 2, \ldots, n$

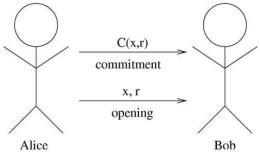
посылаем участнику i пару чисел $(i, s(i))$

Заставим раздающего доказать, что он действительно выдал правильные значения одного фиксированного полинома

Привязка к биту

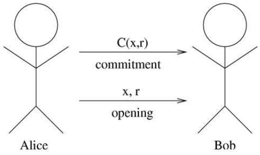


Привязка к биту



Пусть p, q простые числа, $p = 2q + 1$. Мы будем рассматривать подгруппу квадратичных остатков ($a \equiv w^2$) по модулю p.

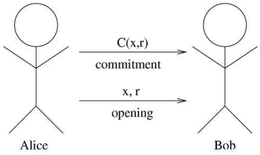
Привязка к биту



Пусть p, q простые числа, $p = 2q + 1$. Мы будем рассматривать подгруппу квадратичных остатков ($a \equiv w^2$) по модулю p.

Факт: По данным g и h очень трудно определить такое x, что $g^x \equiv h(\mod p)$.

Привязка к биту



Пусть p, q простые числа, $p = 2q + 1$. Мы будем рассматривать подгруппу квадратичных остатков ($a \equiv w^2$) по модулю p.

Факт: По данным g и h очень трудно определить такое x, что $g^x \equiv h(\mod p)$.

Построим на основе этого факта **гомоморфную** схему привязки к биту.

Гомоморфная привязка к биту

Подготовка: Боб объявляет $p = 2q + 1$, остатки g и h

Гомоморфная привязка к биту

Подготовка: Боб объявляет $p = 2q + 1$, остатки g и h

Кодируем: $C(x, r) = g^x \cdot h^r$

Гомоморфная привязка к биту

Подготовка: Боб объявляет $p = 2q + 1$, остатки g и h

Кодируем: $C(x, r) = g^x \cdot h^r$

Абсолютная секретность: $\forall y \exists q \, : \, g^x \cdot h^r = g^y \cdot h^q$

Гомоморфная привязка к биту

Подготовка: Боб объявляет $p = 2q + 1$, остатки g и h

Кодируем: $C(x, r) = g^x \cdot h^r$

Абсолютная секретность: $\forall y \exists q \; : \; g^x \cdot h^r = g^y \cdot h^q$

Вычислительная стойкость: нахождение q влечет вычисление дискретного логарифма:
$\log_g h = (x - y)/(q - r) \mod q$

Гомоморфная привязка к биту

Подготовка: Боб объявляет $p = 2q + 1$, остатки g и h

Кодируем: $C(x, r) = g^x \cdot h^r$

Абсолютная секретность: $\forall y \exists q\ :\ g^x \cdot h^r = g^y \cdot h^q$

Вычислительная стойкость: нахождение q влечет вычисление дискретного логарифма:
$\log_g h = (x - y)/(q - r) \mod q$

Гомоморфность:
$C(x, r) \cdot C(y, q) = C(x + y, r + q)$

Старая схема

Подготовительный шаг

Фиксируем простое p и первообразный корень g

Привязка в два шага

Боб выбирает случайное q, посылает Алисе $y = g^q$

Алиса выбирает случайное r, посылает Бобу

$$C(b, r) = y^b g^r$$

Старая схема

Подготовительный шаг

Фиксируем простое p и первообразный корень g

Привязка в два шага

Боб выбирает случайное q, посылает Алисе $y = g^q$

Алиса выбирает случайное r, посылает Бобу

$$C(b, r) = y^b g^r$$

В чем разница (кроме гомоморфности)?

Контроль над раздающим

1. Раздающий случайным образом выбирает коэффициенты $f = x + f_1 z + \cdots + f_{t-1} z^{t-1}$ и $f' = f'_0 + \cdots + f'_{t-1} z^{t-1}$

Контроль над раздающим

1. Раздающий случайным образом выбирает коэффициенты $f = x + f_1 z + \cdots + f_{t-1} z^{t-1}$ и $f' = f'_0 + \cdots + f'_{t-1} z^{t-1}$
2. Раздающий публикует привязки для коэффициентов: $A_i = g^{f_i} \cdot h^{f'_i}$

Контроль над раздающим

1. Раздающий случайным образом выбирает коэффициенты $f = x + f_1 z + \cdots + f_{t-1} z^{t-1}$ и $f' = f'_0 + \cdots + f'_{t-1} z^{t-1}$
2. Раздающий публикует привязки для коэффициентов: $A_i = g^{f_i} \cdot h^{f'_i}$
3. Участнику номер i раздающий посылает $f(i), f'(i)$

Контроль над раздающим

1. Раздающий случайным образом выбирает коэффициенты $f = x + f_1 z + \cdots + f_{t-1} z^{t-1}$ и $f' = f'_0 + \cdots + f'_{t-1} z^{t-1}$
2. Раздающий публикует привязки для коэффициентов: $A_i = g^{f_i} \cdot h^{f'_i}$
3. Участнику номер i раздающий посылает $f(i), f'(i)$
4. Каждый участник выполняет проверку:

$$g^{f(i)} \cdot h^{f'(i)} = A_0 \cdot A_1^{i} \cdot A_2^{i^2} \cdot \cdots \cdot A_{t-1}^{i^{t-1}}$$

Контроль над раздающим

1. Раздающий случайным образом выбирает коэффициенты $f = x + f_1 z + \cdots + f_{t-1} z^{t-1}$ и $f' = f'_0 + \cdots + f'_{t-1} z^{t-1}$
2. Раздающий публикует привязки для коэффициентов: $A_i = g^{f_i} \cdot h^{f'_i}$
3. Участнику номер i раздающий посылает $f(i), f'(i)$
4. Каждый участник выполняет проверку:
$$g^{f(i)} \cdot h^{f'(i)} = A_0 \cdot A_1^{i} \cdot A_2^{i^2} \cdot \cdots \cdot A_{t-1}^{i^{t-1}}$$
5. Если проверка не прошла, участник сообщает всем остальным

Контроль над раздающим

1. Раздающий случайным образом выбирает коэффициенты $f = x + f_1 z + \dots + f_{t-1} z^{t-1}$ и $f' = f'_0 + \dots + f'_{t-1} z^{t-1}$
2. Раздающий публикует привязки для коэффициентов: $A_i = g^{f_i} \cdot h^{f'_i}$
3. Участнику номер i раздающий посылает $f(i), f'(i)$
4. Каждый участник выполняет проверку:
$$g^{f(i)} \cdot h^{f'(i)} = A_0 \cdot A_1^i \cdot A_2^{i^2} \cdot \dots \cdot A_{t-1}^{i^{t-1}}$$
5. Если проверка не прошла, участник сообщает всем остальным

Контроль над раздающим

1. Раздающий случайным образом выбирает коэффициенты $f = x + f_1 z + \cdots + f_{t-1} z^{t-1}$ и $f' = f'_0 + \cdots + f'_{t-1} z^{t-1}$
2. Раздающий публикует привязки для коэффициентов: $A_i = g^{f_i} \cdot h^{f'_i}$
3. Участнику номер i раздающий посылает $f(i), f'(i)$
4. Каждый участник выполняет проверку:
$$g^{f(i)} \cdot h^{f'(i)} = A_0 \cdot A_1^i \cdot A_2^{i^2} \cdot \cdots \cdot A_{t-1}^{i^{t-1}}$$
5. Если проверка не прошла, участник сообщает всем остальным

Участник считает раздающего жуликом, если:

его собственная проверка не прошла

было объявлено о t нарушениях

менее t участников не объявляли о нарушениях

План лекции

Постановка Рабина

- Два участника: передающий **S** и получающий **R**
- **S** посылает бит b по специальному каналу **R**
- С вероятностью $1/2$ **R** получает b
- С вероятностью $1/2$ **R** получает $\#$
- **S** не имеет никакой информации о том, что досталось **R**

Передача 1-из-2

Авторы: Even, Goldreich и Lempel

Передача 1-из-2

Авторы: Even, Goldreich и Lempel

- Два участника: передающий **S** и получающий **R**
- У **S** есть два бита b_0 и b_1
- У **R** есть бит запроса i
- **R** должен получить b_i, но ничего не узнать о b_{1-i}
- **S** не должен узнать i

Схема Рабина $\Rightarrow$ 1-из-2 схема

Какие будут идеи?

Схема Рабина $\Rightarrow$ 1-из-2 схема

Какие будут идеи?

1. **S** посылает **R** по схеме Рабина $3n$ битов $a_1, \ldots, a_{3n}$

Схема Рабина $\Rightarrow$ 1-из-2 схема

Какие будут идеи?

1. **S** посылает **R** по схеме Рабина $3n$ битов $a_1, \ldots, a_{3n}$
2. **R** посылает две непересекающиеся группы индексов $i_1, \ldots, i_n$ и $j_1, \ldots, j_n$

Схема Рабина $\Rightarrow$ 1-из-2 схема

Какие будут идеи?

1. **S** посылает **R** по схеме Рабина $3n$ битов $a_1, \ldots, a_{3n}$
2. **R** посылает две непересекающиеся группы индексов $i_1, \ldots, i_n$ и $j_1, \ldots, j_n$
3. **S** посылает **R** бита $c_0 = a_{i_1} \oplus \cdots \oplus a_{i_n} \oplus b_0$ и $c_0 = a_{j_1} \oplus \cdots \oplus a_{j_n} \oplus b_1$

Схема Рабина $\Rightarrow$ 1-из-2 схема

Какие будут идеи?

1. **S** посылает **R** по схеме Рабина $3n$ битов $a_1, \ldots, a_{3n}$
2. **R** посылает две непересекающиеся группы индексов $i_1, \ldots, i_n$ и $j_1, \ldots, j_n$
3. **S** посылает **R** бита $c_0 = a_{i_1} \oplus \cdots \oplus a_{i_n} \oplus b_0$ и $c_0 = a_{j_1} \oplus \cdots \oplus a_{j_n} \oplus b_1$
4. Вероятность восстановления одновременно b_0 и b_1 экспоненциальна мала

Схема Рабина $\Rightarrow$ 1-из-2 схема

Какие будут идеи?

1. S посылает R по схеме Рабина $3n$ битов $a_1, \ldots, a_{3n}$
2. R посылает две непересекающиеся группы индексов $i_1, \ldots, i_n$ и $j_1, \ldots, j_n$
3. S посылает R бита $c_0 = a_{i_1} \oplus \cdots \oplus a_{i_n} \oplus b_0$ и $c_0 = a_{j_1} \oplus \cdots \oplus a_{j_n} \oplus b_1$
4. Вероятность восстановления одновременно b_0 и b_1 экспоненциальна мала
5. S не может определить, для какой группы индексов R получил все значения

Схема 1-из-2 $\Rightarrow$ схема Рабина

Какие будут идеи?

Схема 1-из-2 $\Rightarrow$ схема Рабина

Какие будут идеи?

1. Нужно передать бит b

Схема 1-из-2 $\Rightarrow$ схема Рабина

Какие будут идеи?

1. Нужно передать бит b
2. **S** случайно выбирает a

Схема 1-из-2 $\Rightarrow$ схема Рабина

Какие будут идеи?

1. Нужно передать бит b
2. **S** случайно выбирает a
3. С вероятностью $1/2$ запускаем 1-из-2 схему для пары (a, b)

Схема 1-из-2 $\Rightarrow$ схема Рабина

Какие будут идеи?

1. Нужно передать бит b
2. **S** случайно выбирает a
3. С вероятностью $1/2$ запускаем 1-из-2 схему для пары (a, b)
4. С вероятностью $1/2$ запускаем 1-из-2 схему для пары (b, a)

Схема 1-из-2 $\Rightarrow$ схема Рабина

Какие будут идеи?

1. Нужно передать бит b
2. **S** случайно выбирает a
3. С вероятностью $1/2$ запускаем 1-из-2 схему для пары (a, b)
4. С вероятностью $1/2$ запускаем 1-из-2 схему для пары (b, a)
5. **R** получает один из битов

Схема 1-из-2 $\Rightarrow$ схема Рабина

Какие будут идеи?

1. Нужно передать бит b
2. **S** случайно выбирает a
3. С вероятностью $1/2$ запускаем 1-из-2 схему для пары (a, b)
4. С вероятностью $1/2$ запускаем 1-из-2 схему для пары (b, a)
5. **R** получает один из битов
6. **S** сообщает какой из вариантов 3 или 4 имел место.

Протокол для 1-из-2

Будем использовать **односторонние перестановки с секретом**

Протокол для 1-из-2

Будем использовать **односторонние перестановки с секретом**

1. **S** сообщает одностороннюю функцию f **R**

Протокол для 1-из-2

Будем использовать **односторонние перестановки с секретом**

1. **S** сообщает одностороннюю функцию f **R**
2. **R** выбирает случайно x_i, считает $y_i = f(x_i)$ и выбирает случайное y_{1-i}

Протокол для 1-из-2

Будем использовать **односторонние перестановки с секретом**

1. **S** сообщает одностороннюю функцию f **R**
2. **R** выбирает случайно x_i, считает $y_i = f(x_i)$ и выбирает случайное y_{1-i}
3. **R** посылает y_0 и y_1 **S**

Протокол для 1-из-2

Будем использовать **односторонние перестановки с секретом**

1. **S** сообщает одностороннюю функцию f **R**
2. **R** выбирает случайно x_i, считает $y_i = f(x_i)$ и выбирает случайное y_{1-i}
3. **R** посылает y_0 и y_1 **S**
4. **S** высылает $b_0 \oplus HCB(f^{-1}(y_0))$ и $b_1 \oplus HCB(f^{-1}(y_1))$

Протокол для 1-из-2

Будем использовать **односторонние перестановки с секретом**

1. **S** сообщает одностороннюю функцию f **R**
2. **R** выбирает случайно x_i, считает $y_i = f(x_i)$ и выбирает случайное y_{1-i}
3. **R** посылает y_0 и y_1 **S**
4. **S** высылает $b_0 \oplus HCB(f^{-1}(y_0))$ и $b_1 \oplus HCB(f^{-1}(y_1))$
5. **R** может восстановить только один бит

Протокол для 1-из-2

Будем использовать **односторонние перестановки с секретом**

1. **S** сообщает одностороннюю функцию f **R**
2. **R** выбирает случайно x_i, считает $y_i = f(x_i)$ и выбирает случайное y_{1-i}
3. **R** посылает y_0 и y_1 **S**
4. **S** высылает $b_0 \oplus HCB(f^{-1}(y_0))$ и $b_1 \oplus HCB(f^{-1}(y_1))$
5. **R** может восстановить только один бит

Протокол для 1-из-2

Будем использовать **односторонние перестановки с секретом**

1. S сообщает одностороннюю функцию f R
2. R выбирает случайно x_i, считает $y_i = f(x_i)$ и выбирает случайное y_{1-i}
3. R посылает y_0 и y_1 S
4. S высылает $b_0 \oplus HCB(f^{-1}(y_0))$ и $b_1 \oplus HCB(f^{-1}(y_1))$
5. R может восстановить только один бит

Как заставить R выбрать y_{1-i} случайным образом?

Форсируем случайность битов

1. R выбирает случайно r_1, посылает $Q = C(r_1, R)$ для S

Форсируем случайность битов

1. R выбирает случайно r_1, посылает $Q = C(r_1, R)$ для S
2. S присылает R другую случайную строку r_2

Форсируем случайность битов

1. R выбирает случайно r_1, посылает $Q = C(r_1, R)$ для S
2. S присылает R другую случайную строку r_2
3. R использует $r_1 \oplus r_2$ в качестве y_{1-i}

Форсируем случайность битов

1. R выбирает случайно r_1, посылает $Q = C(r_1, R)$ для S
2. S присылает R другую случайную строку r_2
3. R использует $r_1 \oplus r_2$ в качестве y_{1-i}
4. R доказывает с **нулевым разглашением**:

Форсируем случайность битов

1. R выбирает случайно r_1, посылает $Q = C(r_1, R)$ для S
2. S присылает R другую случайную строку r_2
3. R использует $r_1 \oplus r_2$ в качестве y_{1-i}
4. R доказывает с **нулевым разглашением**:

Форсируем случайность битов

1. R выбирает случайно r_1, посылает $Q = C(r_1, R)$ для S
2. S присылает R другую случайную строку r_2
3. R использует $r_1 \oplus r_2$ в качестве y_{1-i}
4. R доказывает с **нулевым разглашением**:

$$\exists R \exists r_1 \exists i \ : \ Q = C(r_1, R) \ \& \ y_i = r_1 \oplus r_2$$

План лекции

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?

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

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?

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

Последний слайд

Если не запомните ничего другого:

- Проверяемое разделение секрета основано на гомоморфной привязке к биту

Последний слайд

Если не запомните ничего другого:

- Проверяемое разделение секрета основано на гомоморфной привязке к биту
- Два подхода к передаче данных вслепую: Модель Рабина и 1-из-2

Последний слайд

Если не запомните ничего другого:

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

Последний слайд

Если не запомните ничего другого:

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

Последний слайд

Если не запомните ничего другого:

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

Вопросы?