# Криптография, Лекция № 8

27 октября 2014 г.

## 1 Протоколы идентификации

Примеры: залогиниться на сайте; доказать в банкомате, что вы владелец. Общая задача: тот кто должен может зарегестрироваться, а тот кто не должен не мог. Бывают идентификации на сайте, обычно нужно написать логин и пароль. Хранить пароль открытым текстом на сервере - понятно, что плохо. Можно хранить в закрытом виде, и сравнивать хэш значения (то есть вызывать конкретную программу, иначе злоумышленник может перехватить само хэш значение; а лучше еще и уметь доказывать, что ты пользуешься именно этой программой). Другой тип атак - фальшивые банкоматы, жертва вставляет карточку, вводит пин, ничего не происходит и жертва думает, что просто банкомат сломался. Подобное есть и на сайтах - phishing. Цель построить протокол, который спасет и от поддельных серверов.

### 1.1 Протокол с закрытым ключом

Здесь будет две стороны $P$ - proover, и $V$ - verificator. Через закрытый канал они оба знают $d$ - закрытый ключ. Соответсвенно, есть протокол (вероятностный алгоритм) генерации ключей $K$. $P$, $V$ - интерактивные вероятностные алгоритмы. В конце $V$ выдает 0 или 1.

Условия:

1. Корректность:
$$Pr\{V^{P(d)}(d) = 1\} \simeq 1$$

2. Надежность: Злоумышленник запустил программу $C$ она $k$ раз пообщалась с правильным $P$ после этого она пошла к верификатору и попыталась выступить в качестве прувера. Сделать это ей не должно удасться.
$$\forall C \; \forall k \; Pr\{V^{C^{P^k(d)}}(d) = 1\} \simeq 0$$

Потребуются псевдослучайные функции. Что это такое? Что значит делать запросы к случайной функции? Это значит, что если зафиксированы области значения и определения, то случайно выбирается одна из понятно скольких (двойная экспонента). Псевдослучайных функций хорошо бы чтобы была одинарная экспонента, чтобы можно было записывать номер строкой полиномиальной длины.

**Definition 1.**
ПСФ - семейство $f_e\colon \{0,1\}^n \mapsto \{0,1\}^n$, $e \in \{0,1\}^{k(n)}$.

- Вычислимость: $e, x \mapsto f_e(x)$ - вычисляется полиномиальным алгоритмом.

- Неотличимость: пусть $C_n$ - схемы полиномиального размера с оракульным доступом к $f$ (то есть алгоритмы с подсказкой для всех входов данной длины и оракулом)

$$\forall C_n\ Pr_e\{C_n^{f_e} = 1\} \simeq Pr_f\{C_n^f = 1\}$$

  Идея в том, что эта схема может вычислить значения в полиномиальном числе точек и пытается отличить.

Конструкция: рассматривается генератор псеводослучайных чисел $G\colon \{0,1\}^n \mapsto \{0,1\}^{2n}$. Если $e = \varepsilon$, то $f_e(x) = x$. $G(x) = G_0(x)G_1(x)$ ($G_0(x), G_1(x)$ - два слова длины $n$). Если $|e| = 1$, то $f_e(x) = G_e(x)$. Если $|e| = 2, e = e_1e_2$, то $f_e(x) = G_{e_2}(G_{e_1}(x))$. И так далее.

Закрытый ключ - индекс ПСФ. $P$ говорит $V$ вычисли функцию в такой то точке:

$$P \xleftarrow{x} V$$

$$P \xrightarrow{f_d(x)} V$$

## 1.2 Протокол с открытым ключом

Здесь $P$ знает $d$ закрытый ключ, а $V$ - знает $e$ - открытый ключ, который известен вообще всем.

Условия:

1. Корректность:
$$Pr\{V^{P(d)}(e) = 1\} \simeq 1$$

2. Надежность:
$$\forall C\ \forall k\ Pr\{V^{C^{P^k(d)}(e)}(e) = 1\} \simeq 0$$

Построим протокол на базе функции Рабина: есть некоторый модуль $m = p \cdot q$, где $q, p$ - простые числа из $n$ бит; $(x, m) \mapsto (x^2\ mod\ m, m)$ ($(x, m)$ - закрытый ключ, а $(x^2 mod m, m)$ - открытый).

$$P \xrightarrow{z = y^2\ mod\ m} V$$

$$P \xleftarrow{\alpha \in \{0,1\}} V$$

$$P \xrightarrow{u = y \cdot x^\alpha} V$$

И, наверное, это нужно повторить много раз, ибо $P$ может заранее знать $\alpha$ и подготовиться. Вконце $V$ проверяет, что $u^2 = z \cdot (x^2)^\alpha\ (mod\ m)$. Если взломщик знает $u_0 = y \cdot x^0 = y$ и $u_1 = y \cdot x^1 = y \cdot x$, то он узнаёт $x = \frac{u_1}{u_0}$. Почти навероное при разных запусках будут разные $y$. Если $\alpha = 1$, то он узнает квадрат, но если функция рабина не обратима, то это ему не поможет.

Итуитивно понятно, что надежность эквивалентна двум вещам:

1. $$Pr\{V^{C(e)}(e) = 1\} \simeq 0$$

2. У алгоритма $P$ нулевое разглашение.

Покажем первое (через необратимость (односторонность) функции рабина): пусть $C$ обманывает $V$: $C$ выбирает $z$, после этого запускает её для $\alpha = 0$ и для $\alpha = 1$. $C$ выдает $u_0$ и $u_1$ и тогда $x = \frac{u_1}{u_0}$. Это общая идея, которую нужно далее уточнять.

Про второе: кажется, что в этом случае будет даже статистически нулевое разглашение.