Криптография, Лекция № 11

24 ноября 2014 г.

1 Надежный протокол электронной подписи полиномиального числа слов полиномиальной длины без памяти

Idea 1.
Изначально есть два ключа: закрытый - d и открытый - e. S может подписать пару e_0e_1 ключом d. затем подписать $e_{00}e_{01}$ ключом d_0 и $e_{10}e_{11}$ ключом d_1. За n шагов можно получить 2^n пар ключей. Когда нужна новая подпись, S выбирает случайный из сгенерированных 2^n один ключ и подписывает им сообщение.

Например, выпало 0100, *тогда S пошлет e_0e_1 подписанные ключом d, $e_{00}e_{01}$ - ключом d_0, $e_{010}e_{011}$ - d_{01}, $e_{0100}e_{0101}$ - d_{010}, x - d_{0100}. Итого, подписывание происходит только по одной ветке. Почему мы подписываем пары? Потому, что второй мы уже не сможем отпарвить. e_α генерируются псевдослучайно, номер функции - часть закрытого ключа.*

Формально

Definition 1.
(K, S, V) - протокол надежной подписи одного сообщения произвольной длины. K - генератор ключей, S - подписывающий, V - верификатор. $(\bar{K}, \bar{S}, \bar{V})$ - требуемый протокол.

1. Определим $\bar{K}$:
   $\bar{d} = (d, s)$, $\bar{e} = e$, s - идентификатор псевдослучайной функции из $\{0,1\}^* \mapsto \{0,1\}^{l(n)}$, $e(n)$ - число случайных битов, которые использует K.

2. $\bar{S}$:
   $\bar{S}$ получает $\bar{d}$ и x. Выбирает случайное α длины n. Для всех префиксов $\beta \subset \alpha$ генерирует $(e_{\beta 0}, d_{\beta 0})$ и $(e_{\beta 1}, d_{\beta 1})$ при помощи K и $f_s(\beta 0)$ и $f_s \beta 1$ в качестве случайных битов. Также $e_\varepsilon = e$ и $d_\varepsilon = d$, где через ε обозначено пустое слово.

   $$\bar{S}(\bar{d}, x) = (e_0e_1, S(d, e_0e_1), \ldots, e_{\beta 0}e_{\beta 1}, S(d_\beta, e_{\beta 0}e_{\beta 1}), \ldots, S(d_\alpha, x))$$

3. $\bar{V}$ проверяет подпись естественным образом.

Корректность протокола очевидна. Надежность: с экспоненциально малой вероятностью могут выпасть одинаковые α для разных x-ов. Так что можно считать, что такого не происходит. Далее, можно считать, что ПСФ алгоритм имеет доступ к случайному оракулу то есть все пары (d_α, e_α) генерируются независимо друго от друга алгоритмом K.

Пусть схема C взламывает $(\bar{K}, \bar{S}, \bar{V})$ с вероятностью $\varepsilon > \frac{1}{poly(n)}$. Как использовать эту схему для взлома исходного протокола? C получает e, адаптивно генерирует $x_1, \ldots, x_n$ получает подписи $\bar{s}_1, \ldots, \bar{s}_n$ генерирует x' и подпись s', такие, что $\bar{V}(s', x') = 1$ с вероятностью ε. Будем говорить, что слово α использованное, если $\bar{S}$ в ходе атаки C сгенерировал пару (e_α, d_α). Будем говорить, что α особое, если оно использованное и к тому же, $e_\alpha = e'_\alpha$, где e'_α - элемент S', а ключ d_α не использовался для подписи соответсвующего сообщения из s'.