Курс лекций по теоретической криптографии

Тема 11. **Схемы электронной подписи**

Шокуров А.В.

Определение

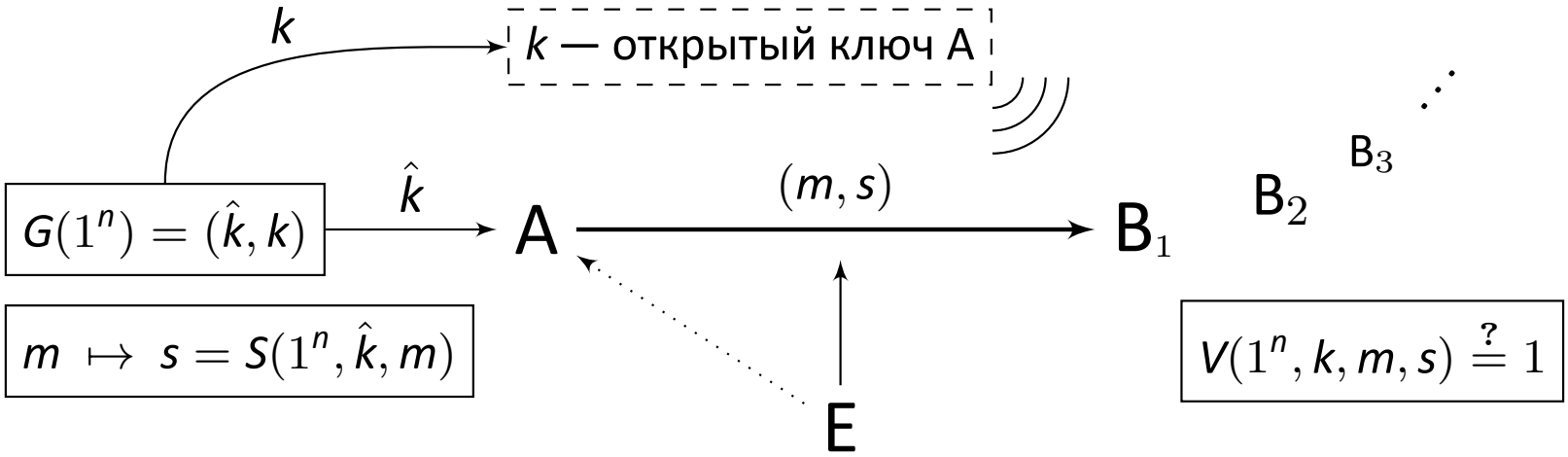
$M_n \subseteq \mathbb{B}^*$ — пространство сообщений

Определение

Схема электронной подписи (electronic / digital signature scheme) — это тройка алгоритмов (G, S, V) вместе с процедурой A, где

1. генератор ключей G (key generation) — п. в. м. Т.:
   $G(1^n) = (\hat{k}, k)$ — секретный и открытый ключи ($K_n = \operatorname{supp} G(1^n)$);
2. генератор подписей S (signing) — п. в. м. Т.:
   $m \in M_n,\ (\hat{k}, k) \in K_n \qquad S(1^n, \hat{k}, m) = s$ — подпись для m;
3. алгоритм проверки V (verification) — п. д. м. Т.:
   $V(1^n, k, m, s) =$
   $$\begin{cases} 1 & (s \text{ принимается как подпись для } m,\ \text{сделанная на ключе } \hat{k}), \\ 0 & (s \text{ отвергается}), \end{cases}$$
   при этом для любых $n \in \mathbb{N},\ m \in M_n,\ (\hat{k}, k) \in K_n$
   $$V\big(1^n, k, m, S(1^n, \hat{k}, m)\big) = 1;$$
4. процедура арбитража A — разрешение споров экспертом (арбитром).

Схема применения и пример



Арбитраж: в спорных случаях, например, когда A отказывается от подписи s для m, вмешивается арбитр, которому участники доверяют и который может запрашивать у A дополнительную информацию: секретный ключ $\hat{k}$ и заполнения «случайных» лент машин G, S.

Схема подписи RSA

$N = pq, \quad p, q \in \mathbb{P}, \qquad (e, \varphi(N)) = 1, \quad ed \equiv 1 \mod \varphi(N)$

- G: $k = (N, e)$, $\hat{k} = (N, d)$
- S: $s = m^d \mod N$
- V: $s^e \stackrel{?}{=} m \mod N \qquad s^e = m^{ed} = m^{1+l\cdot\varphi(N)} = m \cdot 1 \mod N$

MAC'и. Стойкость схем подписи

MAC

Аналогичная схема электронной подписи с секретным ключом (без открытого) — *коды аутентификации сообщений* (message authentication code).

- $G(1^n) = \hat{k}$ известно только участникам A и B.
- Они оба могут создать подпись $s = S(1^n, \hat{k}, m)$. И только они могут проверить её.
- Нет задачи убедить в чём-то третью сторону. MAC предназначен для сети из небольшого числа доверяющих друг другу участников.

Стойкость схемы электронной подписи определяется относительно конкретного противника (против конкретной угрозы на основе конкретной атаки).

Модель противника (adversary model):

- *вычислительные ресурсы* = п. в. м. Т.,
- *атака* (возможности получения исходных данных),
- *угроза* (цель противника).

Противнику известна схема, то есть алгоритмы G, S, V, и параметр стойкости n. Параметр стойкости n и ключи $\hat{k}, k$ во время работы противника не меняются.

Основные типы атак

1. атака с известным открытым ключом (known public key attack, key-only attack, **KKA**);
2. атака с известными сообщениями (known message attack, **KMA**): $(m_1, s_1), (m_2, s_2), \ \ldots, (m_l, s_l)$, где $s_i = S(1^n, \hat{k}, m_i)$;
3. атака с выбором сообщений (chosen message attack, **CMA**): $m_1, \ \ldots, m_l \rightsquigarrow (m_1, s_1), \ \ldots, (m_l, s_l)$, где $s_i = S(1^n, \hat{k}, m_i)$.

$1 \leqslant i \leqslant l$

$l \leqslant \mathrm{poly}(n)$

Варианты атак последнего типа:

- *простая* (generic attack) не направлена на какого-то конкретного участника (m_i выбираются независимо от k, но s_i — их подписи на $\hat{k}$),
- *направленная* (directed attack) на определённого участника (противник знает k изначально);

- *неадаптивная*, когда противник выбирает m_i и получает весь набор s_i сразу,
- *адаптивная*, когда подписи к выбранным сообщениям он получает последовательно по i, то есть выбор m_i может зависеть от результатов предыдущих запросов (то есть от $(m_1, s_1), \ \ldots, (m_{i-1}, s_{i-1})$).

NB Атаки, в которых противник получает набор подписей, предполагают, что, вообще говоря, для каждого нового i вероятностная машина S использует новое заполнение своей «случайной» ленты.

Основные типы угроз

1. полное раскрытие (total breaking):
   найти некоторый $\hat{k}'$, соответствующий k, то есть такой, что $(\hat{k}', k) \in K_n$;
2. универсальная подделка (universal forgery):
   найти эффективный алгоритм S', не требующий $\hat{k}$ и функционально эквивалентный S, то есть
   для любого $m \in M_n \qquad S'(1^n, k, m) = s' \;\Rightarrow\; V(1^n, k, m, s') = 1$;
3. селективная подделка (selective forgery):
   для некоторого $m \in M_n \setminus \{m_1, \ldots, m_l\}$ найти s', так что $V(1^n, k, m, s') = 1$;
4. экзистенциальная подделка (existential forgery):
   найти такую пару (m', s'), $m' \neq m_1, \ldots, m_l$, что $V(1^n, k, m', s') = 1$.

Если никакой эффективный алгоритм не может осуществить угрозу экзистенциальной подделки с существенной вероятностью, то схема электронной подписи называется *EU-стойкой* (от слов existential unforgeability).

- Схема подписи RSA не является EU-KKA-стойкой:
  возьмём произвольное s и вычислим $m = s^e \mod N$ — подписью для m служит s.
- Схема подписи RSA не является UU-CMA-стойкой относительно универсальной подделки: для любого m возьмём $r \in_{\mathcal{R}} \mathbb{Z}_N^*$ и вычислим $m_1 = rm \mod N$, $m_2 = r^{-1} \mod N$, тогда подписью для m служит $s_1 \cdot s_2$, где s_i — подпись для m_i.

Формальное определение EU-CMA-стойкости

Рассмотрим оракул-подписант $\mathcal{O}$, соответствующий атаке с выбором сообщений (CMA): на запросы $m_i \in M_n$ он возвращает такие s_i, что $V(1^n, k, m_i, s_i) = 1$, то есть он просто выполняет $S(1^n, \hat{k}, \cdot)$.

Определение

Схема электронной подписи (G, S, V) — *стойкая против угрозы экзистенциальной подделки на основе (адаптивной) атаки с выбором сообщений* (EU-CMA-стойкая), если для любой п. в. м. Т. A с указанным оракулом $\mathcal{O}$

$$\Pr\left[A^{\mathcal{O}}(1^n, k) = (m, s),\ m \neq m_i,\ V(1^n, k, m, s) = 1\right] = \mathrm{negl}(n).$$

Теорема (Rompel)

Если существует односторонняя функция,

то существует EU-CMA-стойкая схема электронной подписи.

NB В другую сторону утверждение тоже верно: функция f, такая, что $f(r) = k \Leftrightarrow G(r; 1^n) = (\hat{k}, k)$, должна быть односторонней, иначе противник с существенной вероятностью по k сможет находить r и вычислять $\hat{k}$.

Схема Лэмпорта (одноразовая)

Пусть f — односторонняя функция, сохраняющая длину, $f(\mathbb{B}^n) \subseteq \mathbb{B}^n$ для всех n.

Сообщение $m \in M_n = \mathbb{B}^n, \quad m = m^{[1]}m^{[2]} \ldots m^{[n]}$

- $G(1^n)$: $x_i^0, x_i^1 \in_{\mathcal{U}} \mathbb{B}^n \quad \rightsquigarrow \quad y_i^0 = f(x_i^0),\ y_i^1 = f(x_i^1), \qquad 1 \leqslant i \leqslant n$

  $\hat{k} = x_1^0 x_1^1 x_2^0 x_2^1 \ldots x_n^0 x_n^1 \in \mathbb{B}^{2n^2}$

  $k = y_1^0 y_1^1 y_2^0 y_2^1 \ldots y_n^0 y_n^1 \in \mathbb{B}^{2n^2}$
- $S(\hat{k}, m)$: $s_i = x_i^{m^{[i]}}, \quad s = s_1 s_2 \ldots s_n = x_1^{m^{[1]}} x_2^{m^{[2]}} \ldots x_n^{m^{[n]}} \in \mathbb{B}^{n^2}$;
- $V(k, m, s)$: $V(k, m, s) = 1 \quad \Leftrightarrow \quad \forall\, i \in \{1, \ldots, n\} \quad y_i^{m^{[i]}} = f(s_i) = f(x_i^{m^{[i]}})$.

Эта схема EU-CMA$_1$-стойкая — при условии, что противнику доступно только одно обращение к оракулу ($l = 1$). В этом смысле схема *одноразовая* (на каждом секретном ключе можно подписать лишь одно сообщение).

? Покажите, что без условия одноразовости (то есть при $l > 1$) схема не стойкая.

Схема с предварительным хэшированием

Чтобы снять ограничение на длину $|m| = n$, воспользуемся семейством криптографических хэш-функций

$$H = \bigcup_{n \in \mathbb{N}} H_n, \qquad H_n = \{h_{n,d} : \mathbb{B}^{l(n)} \to \mathbb{B}^n\}_{d \in I_n}.$$

К ключам добавляется $d \leftarrow \mathcal{I}_n$, а вместо $m \in \mathbb{B}^{l(n)}$ подписывается $h_{n,d}(m) \in \mathbb{B}^n$ (hash-and-sign paradigm). При проверке также сообщение сначала хэшируется.

Преобразованная схема Лэмпорта (G^H, S^H, V^H) для $M_n = \mathbb{B}^{l(n)}$

- $G^H(1^n)$: $(\hat{k}, k) = G(1^n)$
  $d \leftarrow \mathcal{I}_n$
  выход $\big((\hat{k}, d), (k, d)\big)$
- $S^H(\hat{k}, d, m)$: $s = S(\hat{k}, h_{n,d}(m))$
  выход s
- $V^H(k, d, m, s)$: выход $V(\hat{k}, h_{n,d}(m), s)$

Далее будем считать, что хэширование неявно осуществляется каждый раз перед подписыванием сообщения и проверкой подписи. То есть рассматриваемое далее сообщение m — на самом деле его хэш-значение $h_{n,d}(m)$.

Схема многоразовая, с внутренними состояниями

Для каждого нового m_i заранее генерируется новая пара ключей:
когда будет подписываться m_{i-1}, подпишем и открытый ключ для следующего сообщения

$$s_{i-1} = S(\hat{k}_{i-1}, m_{i-1}k_i)$$

Таким образом, $\hat{k}_i$ используются по одному разу, действительность k_i устанавливается проверкой подписей цепочки предыдущих сообщений.

- $G^Q(1^n)$: $(\hat{k}, k) = G(1^n)$

  $\hat{k}_1 = \hat{k}, \quad M = \varnothing, \quad q = (\hat{k}_1, M)$ (q — текущее состояние)
- $S^Q(\hat{k}, m_i \mid q)$: $(\hat{k}_{i+1}, k_{i+1}) = G(1^n)$
  $s_i = S(\hat{k}_i, m_i k_{i+1})$, $\hat{k}_i$ берётся из q
  выход (s_i, M)
  $M = M \cup \{m_i k_{i+1} s_i\}, \quad q = (\hat{k}_{i+1}, M)$
- $V^Q(k, m, s, M)$: $k_1 = k, \quad i = |M| + 1, \quad m_i k_{i+1} = m, \quad s_i = s, \quad v = 1$ (истина)
  $\{m_1 k_2 s_1, \ \ldots, m_{i-1} k_i s_{i-1}\} = M \quad \rightsquigarrow \quad k_1, k_2, \ \ldots, k_i$
  для каждого $j \in \{1, \ \ldots, i\}$:
  $v = v \wedge V(k_j, m_j k_{j+1}, s_j)$
  выход v

то есть $V^Q(k, m, s, M) = 1 \quad \Leftrightarrow \quad \forall j \in \{1, \ \ldots, i\} \quad V(k_j, m_j k_{j+1}, s_j) = 1$

Схема многоразовая, «древовидная»

Будем создавать пару ключей для каждого возможного префикса $x = m^{[1 \ldots i]}$ сообщения m начиная с пустого префикса λ, $\quad 0 \leqslant i \leqslant n$.

$\hat{k}_\lambda = \hat{k}$

$k_\lambda = k$

$(\hat{k}_x, k_x) = G(1^n)$

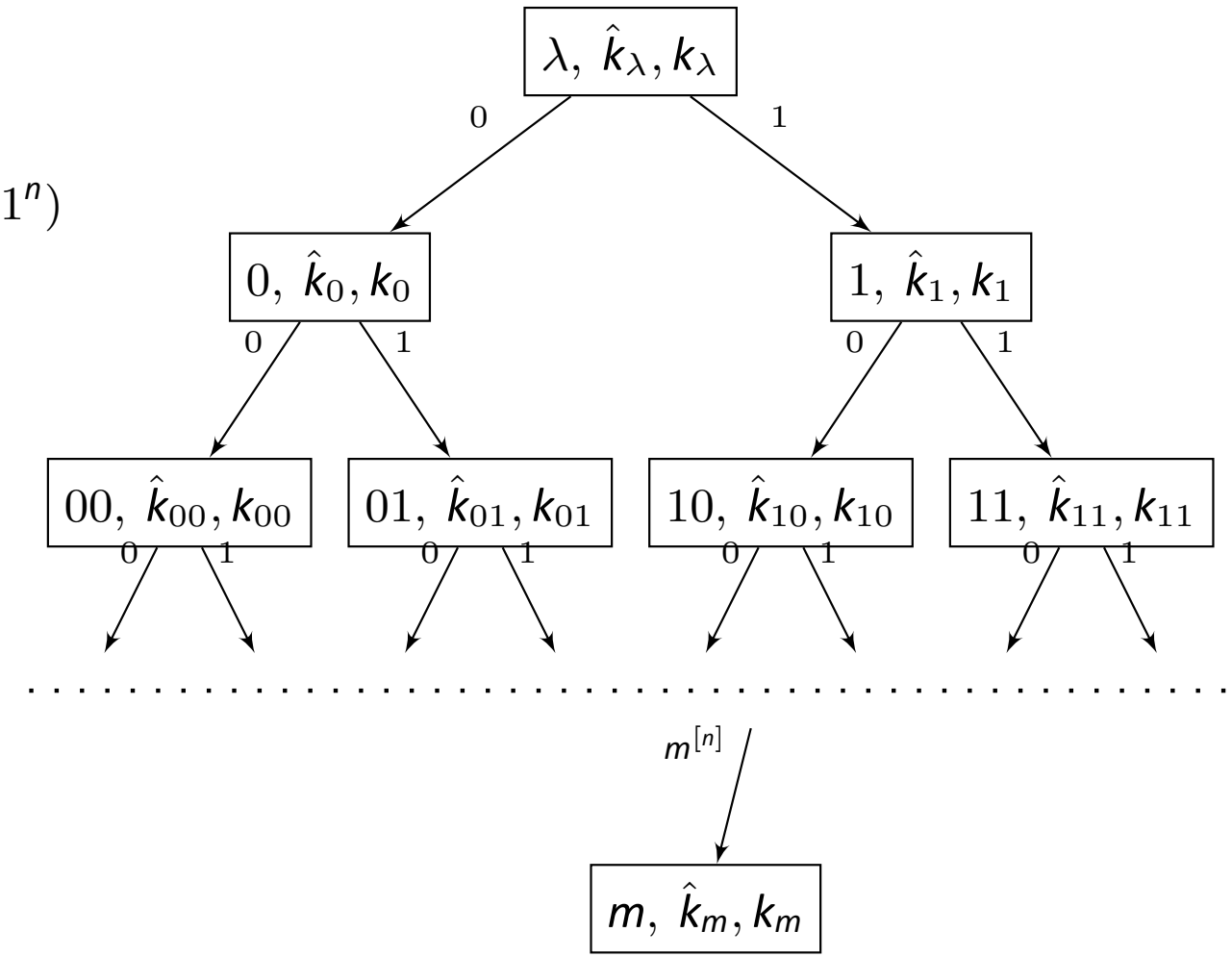


Схема многоразовая, «древовидная»

- $G^T(1^n)$: $(\hat{k}, k) = G(1^n)$

  $\hat{k}_\lambda = \hat{k}, \qquad \forall x \in \mathbb{B}^{\leqslant n} \quad q_x = Null$

- $S^T(\hat{k}, m \mid q)$: $\Sigma = \varnothing$

  для каждого $i \in \{0, \ldots, n\}$:

  $x = m^{[1 \ldots i]}$

  если $q_x = Null$:

  (т. е. этот x ещё не встречался)

  $(\hat{k}_{x0}, k_{x0}) = G(1^n)$

  $(\hat{k}_{x1}, k_{x1}) = G(1^n)$

  $s_x = S(\hat{k}_x, xk_{x0}k_{x1})$

  $q_x = (s_x,\ \hat{k}_{x0},\ \hat{k}_{x1},\ xk_{x0}k_{x1})$

  $\Sigma = \Sigma \cup \{(s_x, k_{x0}, k_{x1})\}$

  выход Σ

- $V^T(k, m, \Sigma)$: $k_\lambda = k, \quad v = 1$ (логическая истина)

  для каждого $i \in \{0, \ldots, n\}$:

  $x = m^{[1 \ldots i]}$, найти (s_x, k_{x0}, k_{x1}) в Σ

  $v = v \wedge V(k_x,\ xk_{x0}k_{x1},\ s_x)$

  выход v

Схема многоразовая, без состояний

Возьмём псевдослучайное семейство функций

$$F = \bigcup_{n\in\mathbb{N}} F_n = \{f_{n,t} : \mathbb{B}^{l(n)} \to \mathbb{B}^{m(n)}\}_{\substack{n\in\mathbb{N}\\ t\in\mathbb{B}^n}},$$

то есть полиномиально вычислимое семейство функций, такое, что для любой п. в. м. Т. A

$$\left|\Pr_t\left[A^{f_{n,t}}(1^n) = 1\right] - \Pr_\varphi\left[A^{\varphi}(1^n) = 1\right]\right| = \mathrm{negl}(n),$$

где $t \in_{\mathcal{U}} \mathbb{B}^n$, $\varphi \in_{\mathcal{U}} \mathrm{Fun}(\mathbb{B}^{l(n)}, \mathbb{B}^{m(n)})$.

С его помощью можно «усовершенствовать» предыдущую схему подписи: вместо того, чтобы помнить состояния для всех префиксов сообщений, перегенерируем их, когда будет нужно. При этом случайную ленту генератора подписей надо будет заполнять одинаковым образом. Для этого и используется псевдослучайное семейство функций F.

Отличия от предыдущей схемы:

секретный ключ $(\hat{k}, t)$, где $t \in_{\mathcal{U}} \mathbb{B}^n$

$s_x = S(r;\ \hat{k}_x,\ xk_{x0}k_{x1}), \quad r = f_{n,t}(x)$

Все приведённые схемы EU-CMA-стойкие в соответствующих предположениях.