Курс лекций по теоретической криптографии

Тема 14. **Системы электронных платежей**

Шокуров А.В.

Система электронных платежей — это совокупность протоколов, реализующих три основные транзакции:

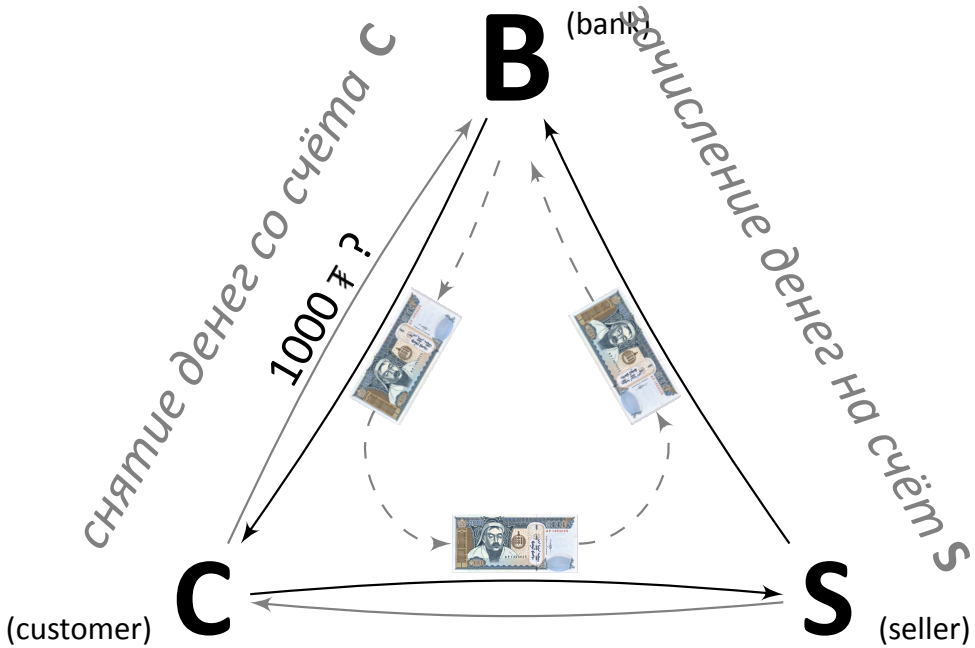
- снятие клиентом денег с его счёта в банке,
- платёж (перевод денег между двумя счетами),
- зачисление денег на счёт в банке.

«Криптографические» задачи, решаемые такой системой, — обеспечение *конфиденциальности*, *целостности* и *неотслеживаемости*.

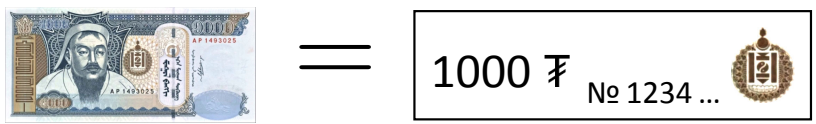
«Электронные наличные» (e-cash) — замена бумажных купюр и монет.
Они представлены двоичными строками (из $\mathbb{B}^*$) $\Longrightarrow$ легко копируются.

NB Обычные банковские системы с пластиковыми картами, онлайн-платежами и т. д. сюда не относятся. Они используют традиционные способы перевода денег между счетами с оформлением платёжных поручений в электронном виде. Основное внимание, таким образом, уделяется аутентификации клиентов банка и передаваемых сообщений (*целостность*). Неотслеживаемость не обеспечивается!

Общая схема



платёж покупателя продавцу



C и **S** держат деньги на счетах в банке **B**, которому они доверяют (но не во всём).

Транзакции

Снятие денег клиентом **C** со своего счёта в банке **B**

- **C** запрашивает у банка **B** банкноту номиналом v.
- **B** проверяет личность **C** (процедура аутентификации) и наличие на счету **C** достаточных средств.
- **B** выдаёт клиенту **C** требуемую электронную банкноту $m = (v, a)$ и списывает сумму v с его счёта. (a — некоторая дополнительная информация о банкноте.)

Платёж покупателя / клиента **C** продавцу **S** за товар / услугу

- **C** передаёт продавцу **S** электронную банкноту m номиналом v.
- **S** проверяет, что банкнота m не поддельная и что она не была потрачена ранее.
- **S** передаёт товар / оказывает услугу **C**.

NB Случай, когда продавец должен вернуть покупателю сдачу, здесь не рассматривается. Отдельной проблемы это не составляет.

Зачисление денег на счёт **S** в банке **B**

- **S** передаёт банку **B** банкноту m номиналом v.
- **B** проверяет, что банкнота m не поддельная и что она не была принята им ранее.
- **B** пополняет счёт **S** на сумму v и запоминает банкноту m как использованную.

Требования к системе

Специфические требования к системе электронных платежей
(помимо обычных — конфиденциальности и целостности системы)

1. *неподделываемость* электронных банкнот (unforgeability):
   нельзя из l банкнот сделать $(l+1)$-ую (без участия банка);
2. *неотслеживаемость* (anonymity: untraceability, unlinkability ...):
   нельзя связать конкретных пользователей с определёнными действиями, например, нельзя отследить, кто из клиентов банка потратил свои банкноты на приобретение некоторого товара у определённого продавца
   (необходимо большое количество пользователей и банкнот каждого достоинства);
3. *невозможность повторной траты* одной банкноты (double-spending prevention):
   нельзя незаметно (безнаказанно) расплатиться дважды одной и той же электронной банкнотой.

NB Кроме алгоритмов (для осуществления транзакций), обеспечивающих выполнение данных требований, нужны ещё процедуры арбитража (на случай отказа от подписи, споров о состоянии счёта в банке, о недошедшей банкноте и т. д.), но мы их здесь рассматривать не будем.

Снятие денег со счёта и неподделываемость банкнот

Решение задачи неподделываемости — электронная подпись банка на банкноте.

Транзакция снятия денег (получения банкнот)

- **Клиент C доказывает свою аутентичность банку B**
- **C** создаёт банкноту (заготовку) $m = (v, a)$ и отправляет её банку **B**
- **B** проверяет m и состояние счёта **C**,
  подписывает m с помощью схемы электронной подписи,
  возвращает (m, s_m^{B}) клиенту **C** и списывает сумму v со счёта **C**

$\Rightarrow$ действительная электронная банкнота — $\big((v, a), s_m^{\mathsf{B}}\big)$, где $m = (v, a)$.

Пример: схема подписи RSA

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

- e — открытый ключ банка **B**, d — его соответствующий секретный ключ
- подпись: $s_m^{\mathsf{B}} = m^d \mod N$
- проверка подписи: $(s_m^{\mathsf{B}})^e \stackrel{?}{=} m \mod N$

Проблема: банк может отследить путь банкноты, а клиент может её размножить.

Неотслеживаемость

Решение задачи неотслеживаемости — использование *схемы электронной подписи вслепую* (blind signature):

C даёт на подпись банку вместо заготовки m сообщение m' так, чтобы из подписи для m' можно было восстановить подпись для m, но банк при этом ничего не узнал об m.

Схема подписи вслепую на основе RSA

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

- e — открытый ключ банка **B**, d — его соответствующий секретный ключ
- **C** выбирает случайный элемент r из $\mathbb{Z}_N^*$, r^e — затемняющий множитель
- подпись банка для $m' = m \cdot r^e \mod N$: $s^{\mathsf{B}}_{m'} = (m')^d \mod N$
- **C** вычисляет подпись для m: $s^{\mathsf{B}}_m = s^{\mathsf{B}}_{m'} \cdot r^{-1} \mod N$
- проверка подписи: $(s^{\mathsf{B}}_m)^e \stackrel{?}{=} m \mod N \qquad (\Leftarrow \quad s^{\mathsf{B}}_{m'} = m^d \cdot r \mod N)$

NB Для предотвращения криминального оборота электронных банкнот можно использовать т. н. *законную* схему подписи вслепую (fair blind signature) — что-то вроде подписи с секретом, который доступен арбитру и который позволяет отследить действия клиентов банка.

Проблема: банк не видит, что подписывает, и поэтому клиент может подсунуть ему, например, банкноту с номиналом, большим заявленного.

Предотвращение подписания «неправильной» банкноты

Возможные **решения**

- **C** должен доказать банку правильность заготовки m (например, с помощью интерактивного доказательства с нулевым разглашением).
- У банка для каждого номинала банкнот имеется своя пара *открытый ключ / секретный ключ*, так что успешная проверка подписи банка на определённом открытом ключе гарантирует, что банкнота имеет конкретный номинал:

  B: $(e_1, d_1),\ (e_2, d_2),\ (e_5, d_5),\ \ldots,\ (e_v, d_v),\ \ldots,\ (e_{1000}, d_{1000})$

  S: $V(e_v, m, s^{\mathsf{B}}_m) = 1 \quad \Rightarrow \quad$ номинал m равен v.
- **C** создаёт одновременно t заготовок банкнот одного номинала и затемняет их, банк выбирает наугад одну из них и подписывает её вслепую, потом требует раскрыть все заготовки и проверяет, что все они имеют заявленный номинал:

  C: $m_1 = (v, a_1),\ \ldots, m_t = (v, a_t) \quad \rightsquigarrow \quad m'_1,\ \ldots, m'_t$

  C $\to$ **B**: $m'_1,\ \ldots, m'_t$

  B: $i \in_{\mathcal{U}} \{1,\ \ldots, t\}, \quad m'_i \rightsquigarrow s^{\mathsf{B}}_{m'_i}$

  C $\to$ **B**: $(v, a_1),\ \ldots, (v, a_{i-1}), (v, a_{i+1}),\ \ldots, (v, a_t)$

  B $\to$ **C**: $s^{\mathsf{B}}_{m'_i}$

  Вероятность необнаружения банком мошенничества со стороны **C** равна $\frac{1}{t}$.

Защита от повторной траты одной банкноты

Решения

- *Централизованные* системы (online):

  B ведёт реестр поступивших к нему (использованных) банкнот,
  - **S**, получив (m, s_m^B) от **C**, запрашивает банк,
  - **B** проверяет нет ли этой банкноты в реестре,
  - если нет, то деньги зачисляются на счёт **S**,
  - **S** передаёт товар **C**.
- *Автономные* системы (offline):

  S принимает банкноты от **C** без участия банка, но в случае повторной траты при последующем обращении к банку мошенник будет идентифицирован (тогда с него можно будет взыскать ущерб).

Последнее можно обеспечить с помощью т. н. *случайной идентификационной строки* (random identity string, **RIS**), которая вырабатывается покупателем и продавцом при выполнении транзакции (протокола) платежа.

RIS должна обладать свойствами:

- в каждой новой транзакции платежа RIS отличается от всех предыдущих,
- только **C** может создать корректную RIS,
- две различные RIS для одной и той же банкноты (когда покупатель жульничает) позволяет банку идентифицировать покупателя.

Пример системы электронных платежей: снятие денег со счёта

H — универсальное одностороннее семейство хэш-функций.

$[\mathrm{c}]$ — некий идентификатор клиента **C**, известный только ему самому́ и банку **B**.

- **Протокол снятия денег со счёта в банке** (withdrawal protocol)
  1. **C** генерирует t банкнот одного достоинства:

$$m_i = (v, n_i, i, y^0_{i,1}, y^1_{i,1}, \ \ldots, y^0_{i,l}, y^1_{i,l}),$$

     где $y^b_{ij} = h(x^b_{ij})$ для $h \in H$, $(x^0_{ij}, x^1_{ij}) \in_{\mathcal{U}} \{(\alpha, \beta) \mid \alpha \oplus \beta = [\mathrm{c}]\}$,
     $b \in \{0, 1\}$, $1 \leqslant i \leqslant t$, $1 \leqslant j \leqslant l$
  2. **C** затемняет m_i: $m'_i = f(m_i, r_i)$ для случайных r_i — и отправляет m'_i банку
  3. **B** выбирает случайное $i_0 \in_{\mathcal{U}} \{1, \ \ldots, t\}$ и просит раскрыть остальные банкноты
  4. **C** посылает банку m_i, r_i и (x^0_{ij}, x^1_{ij}) для всех $i \neq i_0$ и всех j
  5. **B** проверяет, что во всех $(t-1)$ раскрытых банкнотах
     - номинал равен v
     - $y^0_{ij} = h(x^0_{ij}), \quad y^1_{ij} = h(x^1_{ij})$
     - $x^0_{ij} \oplus x^1_{ij} = [\mathrm{c}]$

     для всех $i \neq i_0$ и всех j
  6. в случае успешной проверки **B** подписывает m'_{i_0}, то есть выдаёт $(m'_{i_0}, s^{\mathsf{B}}_{m'_{i_0}})$
  7. **C** восстанавливает подпись $s^{\mathsf{B}}_{m_{i_0}}$ для банкноты m_{i_0}

Пример системы электронных платежей: платёж и зачисление на счёт

- **Протокол платежа** (payment protocol) [далее опускаем индекс i_0]
  1. **C** передаёт (m, s^{B}_m) продавцу **S**
  2. **S** проверяет подпись s^{B}_m для m
  3. если s^{B}_m корректна, то **S** выбирает случайные $b_1, \ \ldots, b_l \in_{\mathcal{U}} \mathbb{B}$ и отправляет их покупателю **C**
  4. **C** раскрывает $x_j^{b_j} = x_{i_0,j}^{b_j}$ для всех j: $R_m = x_1^{b_1} \ \ldots x_l^{b_l}$ — это RIS
  5. **S** проверяет, что $y_j^{b_j} = h(x_j^{b_j})$ для всех j
  6. в случае успешной проверки **S** принимает банкноту m

- **Протокол зачисления денег на счёт в банке** (deposit protocol)
  1. **S** передаёт (m, s^{B}_m) и R_m банку **B**
  2. **B** проверяет подпись s^{B}_m для m и возможные предыдущие траты m
  3. если всё в порядке, то **B** принимает банкноту m и заносит (m, R_m) в реестр

Если $R_m \neq R'_m = x_1^{b'_1} \ \ldots x_l^{b'_l}$ для m, то $\exists j: \ b_j \neq b'_j \ \Rightarrow$ **B** узнает $x_j^0 \oplus x_j^1 = [\mathsf{c}]$.

Если у **B** оказалось две m с одинаковым R_m, значит, мошенничает продавец **S**.

Даже если кто-то найдёт (m, s^{B}_m), он не сможет выполнить пп. 4–5 —
преимущество перед бумажными деньгами!