

Прикладная Криптография: Симметричные криптосистемы Абсолютная и Семантическая стойкость (Акт 2)

Макаров Артём
МИФИ 2025



Плейлист в ожидании пары

Тест.

- Положить телефон экраном вниз справа от себя
- Не разговаривать с соседями
- Не пользоваться конспектами и электронными устройствами
- Написать номер (по таблице) и ФИО на листочке
- Написать краткий ответ на вопрос
- Дождаться окончания теста



TIME IS
UP

На прошлой лекции...

Эквивалентные определения абсолютной стойкости

Теорема 1.4. Пусть $E = (E, D)$ - шифр Шеннона на (K, M, C) . Рассмотрим вероятностный эксперимент для равномерно распределённой $\mathbf{k} \in_R K$.

Тогда E – абсолютно стойкий тогда и только тогда, когда для произвольного предиката $\phi: C \rightarrow \{0,1\}$ и $\forall m_0, m_1 \in M$

$$\Pr[\phi(E(\mathbf{k}, m_0)) = 1] = \Pr[\phi(E(\mathbf{k}, m_1)) = 1]$$

Иными словами: при использовании произвольного предиката на шифртекстах абсолютно стойкого шифра злоумышленник не получает информации об открытом тексте.

Плохие новости

Теорема 1.7 (Шеннона). Пусть $E = (E, D)$ шифр Шеннона на (K, M, C) . Если E – абсолютно стойкий, то

- $|K| \geq |M|$
- $H(\mathbf{k}) \geq H(\mathbf{m}), \mathbf{k} \in_R K, \mathbf{m} \in_R M$

Простое объяснение – невозможно получить равномерно распределённую случайную величину длины m , используя детерминированный алгоритм над равномерно распределённой случайной величиной длины $n < m$.

Иными словами, для шифрования 1 Gb данных **любым** абсолютно стойким шифром потребуется ключ размера как минимум 1 Gb.

Вычислимый шифр

Вычислимый шифр на (K, M, C) – пара **эффективных** алгоритмов $E = (E, D)$, где $E: K \times M \rightarrow C$ – вероятностная функция зашифрования, $D: K \times C \rightarrow M$ – функция расшифрования.

- Обозначим процедуры зашифрования как $c \stackrel{R}{\leftarrow} E(k, m)$.
- Обозначим выбор **равномерно распределённого ключа** как $k \stackrel{R}{\leftarrow} K$.

При этом $\forall k \in K, m \in M, c \stackrel{R}{\leftarrow} E(k, m), m' \leftarrow D(k, c) \Pr[m = m'] = 1$
(**свойство корректности**).

Семантическая стойкость

Пусть $E = (E, D)$ - вычислимый шифр на (K, M, C) .

Теорема 1.3 $\Rightarrow E$ – абсолютно стойкий, если $\forall \phi: C \rightarrow \{0,1\}, \mathbf{k} \in_R K$ – равномерно распределённый и выполняется

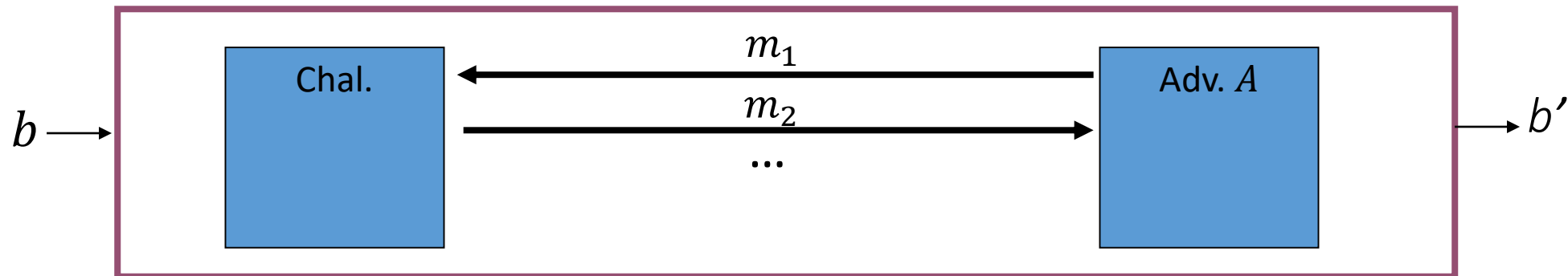
$$\Pr[\phi(E(\mathbf{k}, m_0)) = 1] = \Pr[\phi(E(\mathbf{k}, m_1)) = 1]$$

Ослабим свойство абсолютной стойкости: вместо требования равенства вероятностей потребуем чтобы их разность не превосходила величину ϵ :

$$|\Pr[\phi(E(\mathbf{k}, m_0)) = 1] - \Pr[\phi(E(\mathbf{k}, m_1)) = 1]| \leq \epsilon$$

Понятие игры

- Игра состоит из двух сторон – **противника A (Adversary)** и **претендента (Challenger)**, моделируемые **эффективными** алгоритмами. При этом алгоритм A – вероятностный
- **Входом** игры называется некоторая величина b
- **Ход игры** – атакующий и претендент обмениваются сообщениями согласно некоторому фиксированному протоколу
- **Результатом** игры называется некоторая величина b'

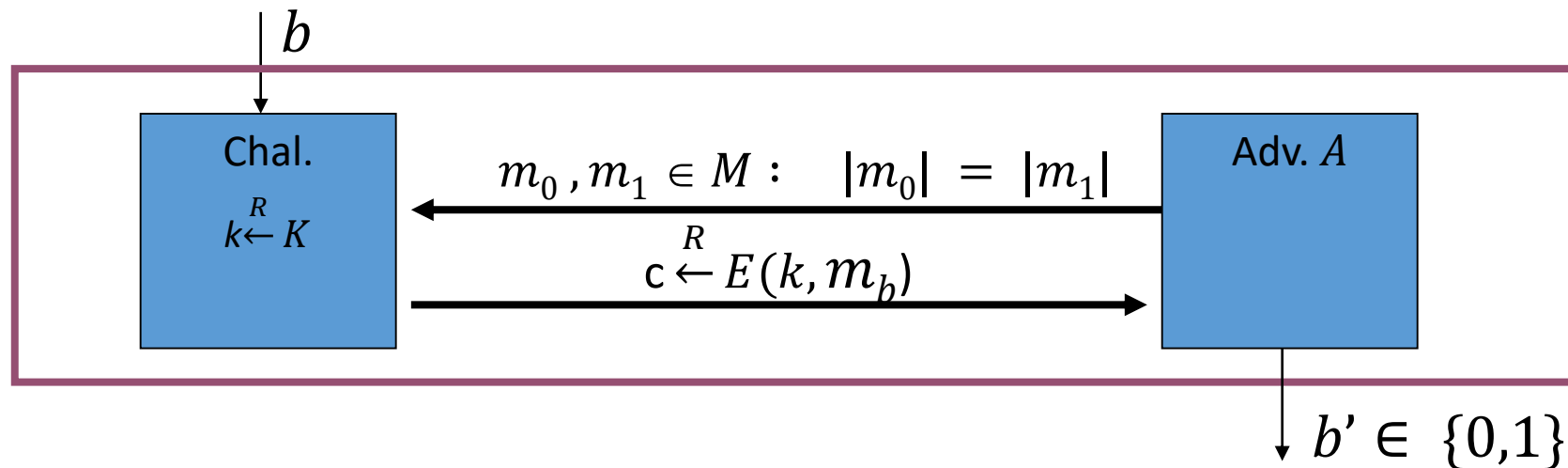


Понятие игры на различимость, определения

- **Входом** игры называется случайное число $b \in \{0,1\}$, неизвестное для атакующего, определяющего эксперимент
- **Экспериментом** ($\text{Exp } b$) называется «режим» претендента при его общении с атакующим
- **Ход игры** – атакующий и претендент обмениваются сообщениями согласно некоторому фиксированному протоколу
- **Цель игры** – атакующий пытается различить два эксперимента
- **Результатом** игры называется число $b' \in \{0,1\}$ – выход алгоритма A

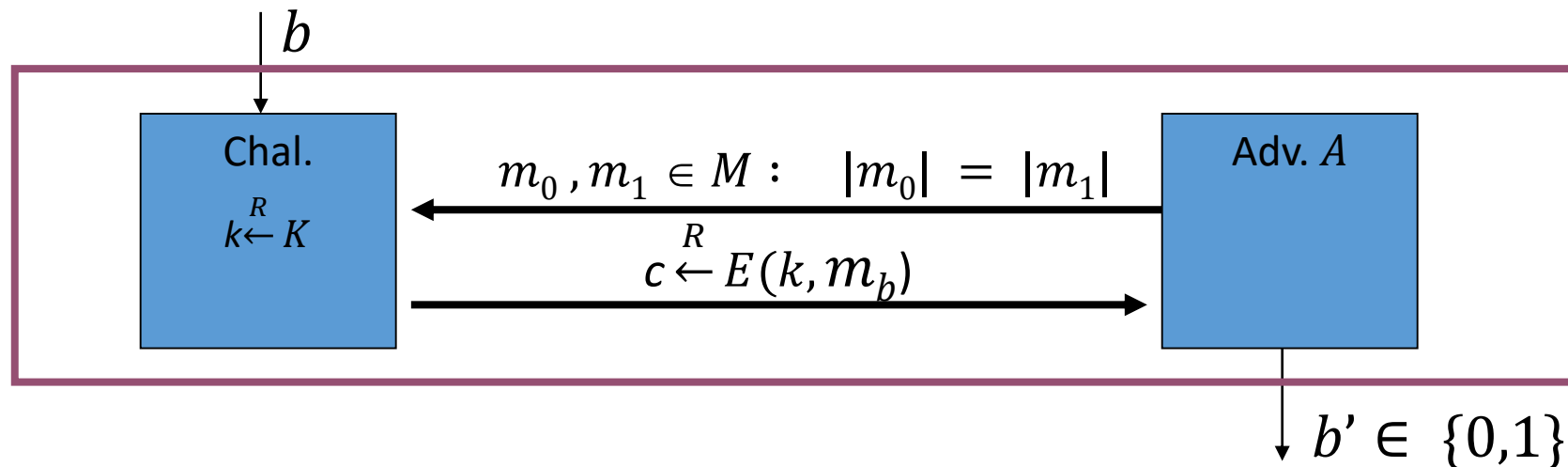
Игра: семантическая стойкость (одноразовое использование ключа)

Для $E = (E, D)$ - вычислимого шифра на (K, M, C) и противника A определим два эксперимента, Exp. 0 и Exp. 1 следующим образом:



Игра: семантическая стойкость (одноразовое использование ключа)

- Претендент выбирает $k \stackrel{R}{\leftarrow} K$
- Противник выбирает сообщения $m_0, m_1 \in M$ **одинаковой длины**
- Претендент вычисляет $c \stackrel{R}{\leftarrow} E(k, m_b)$ и отправляет противнику
- Противник возвращает бит $b' \in \{0,1\}$ как результат игры

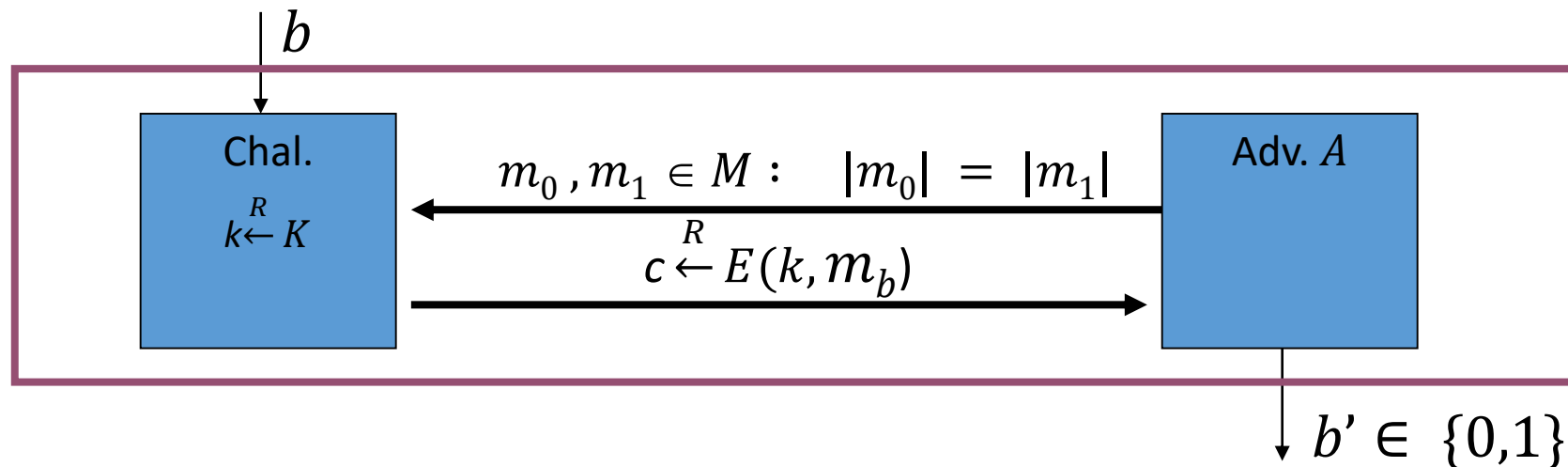


Игра: семантическая стойкость (одноразовое использование ключа)

Пусть W_b - событие того, что $b' = 1$ в эксперименте b .

Преимуществом (Advantage) противника A против алгоритма E в игре на семантическую стойкость есть величина:

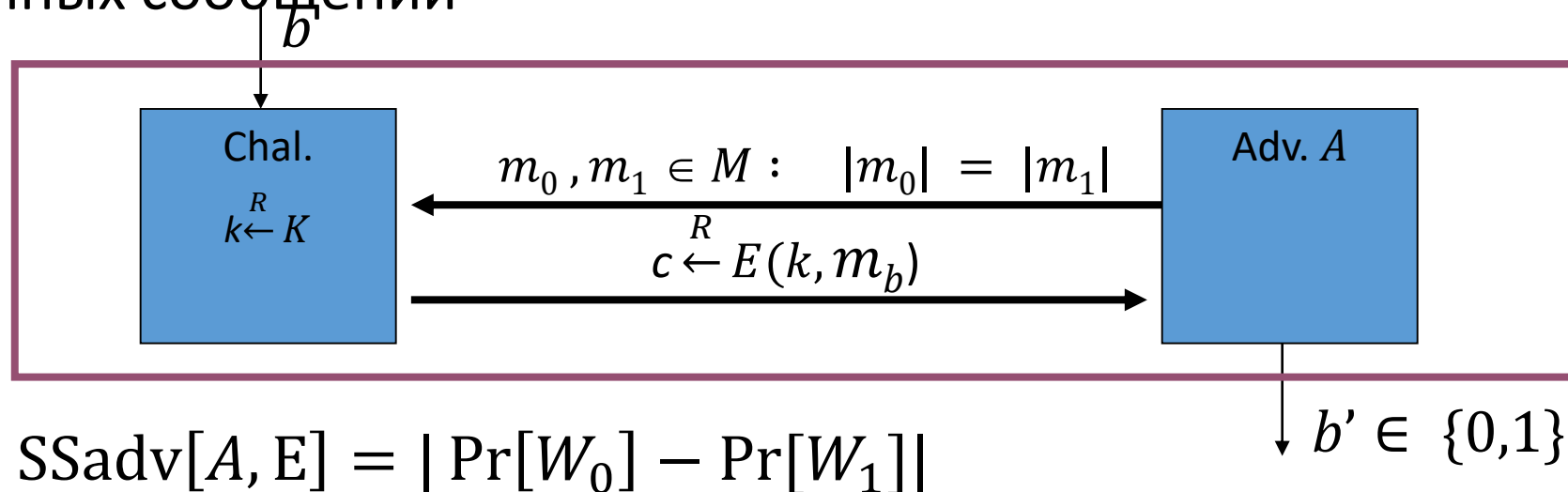
$$\text{SSadv}[A, E] = |\Pr[W_0] - \Pr[W_1]|$$



Семантическая стойкость (одноразовое использование ключа)

Шифр E - (одноразово) **семантически стойкий**, если для всех эффективных противников A величина $SSadv[A, E] < \epsilon$ – **пренебрежимо малая величина**

Иными словами – вычислительно невозможно отличить шифртексты различных сообщений



Семантическая стойкость (одноразовое использование ключа)

Заметим, что нам важна только разница вероятностей событий.

- Большая разность означает, что противник может различать эксперименты
- Малая – что противник различает эксперименты с трудом

Также замети, при замене b' на $b'' = \begin{cases} 0, & b' = 1 \\ 1, & b' = 0 \end{cases}$ как выхода

противника, то его преимущество не изменится.

Иными словами – важна не величина, которая выдаёт противник, а то, что данные величины позволяют различать эксперименты.

Семантическая стойкость

- «Ослабленная» версия абсолютной стойкости: только **эффективные противники** и разность вероятностей расшифрования в заданные сообщения **не превосходит ϵ** .
- Позволяет использовать **короткие ключи**

Примеры:

- Одноразовый блокнот – семантически стойкий шифр
- Одноразовый блокнот переменной длины – семантически стойкий шифр
- Шифр подстановки – не семантически стойкий шифр

Построение атаки на семантическую стойкость

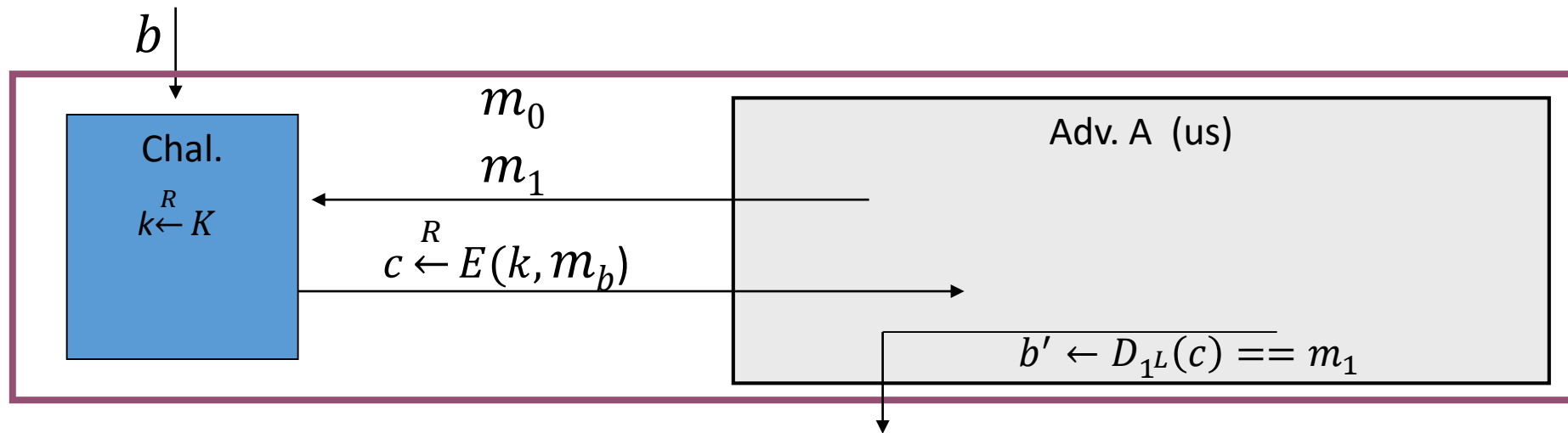
Пусть $E = (E, D)$ – семантически стойкий шифр, $E' = (E', D')$: $E'(k, m) = E(1^L, m)$. Тогда E' - не семантически стойкий.

▷ Построим эффективный алгоритм A , позволяющий выиграть игру на семантическую стойкость.

- Генерация двух различных сообщений m_0, m_1
- Получение шифртекста c для одного из сообщений
- Выдать результат $c == E(1^L, m_1) \triangleleft$

Построение атаки на семантическую стойкость

Пусть $E = (E, D)$ – семантически стойкий шифр, $E' = (E', D')$: $E'(k, m) = E(1^L, m)$. Тогда E' – не семантически стойкий.



$$\text{SSadv}[A, E] = |\Pr[W_0] - \Pr[W_1]| = |1 - 0| = 1$$

Построение атаки на семантическую стойкость (через существующую атаку)

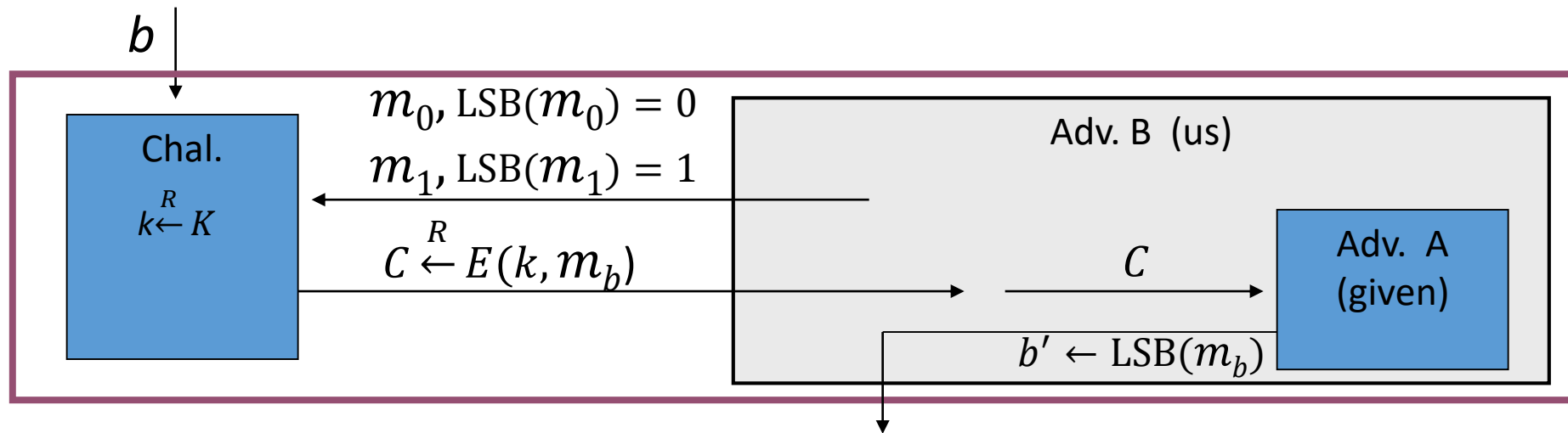
Пусть A – алгоритм позволяющий получить наименее значимый бит (LSB) открытого текста через шифртекст $c \xleftarrow{R} E(k, m)$. Тогда $E = (E, D)$ на (K, M, C) , $M = \{0,1\}^L$ – не семантически стойкий шифр.

▷ Построим эффективный алгоритм B , позволяющий выиграть игру на семантическую стойкость.

- Генерация двух сообщений m_0, m_1 с различным наименее значимым битом
- Получение шифртекста c для одного из сообщений
- Передача шифртекста на вход алгоритма A
- Получение наименее значимого бита открытого текста, определение эксперимента. ◁

Построение атаки на семантическую стойкость (через существующую атаку)

Пусть A – алгоритм позволяющий получить наименее значимый бит (LSB) открытого текста через шифртекст $c \xleftarrow{R} E(k, m)$. Тогда $E = (E, D)$ – не семантически стойкий шифр.



$$\text{SSadv}[B, E] = |\text{Pr}[W_0] - \text{Pr}[W_1]| = |1 - 0| = 1$$

Доказательства сведением (Reduction proof)

Пусть $E = (E, D)$ – вычислимый семантически стойкий шифр на (K, M, C) . Тогда $E' = (E', D')$:
$$\begin{cases} (c_0, c_1) = E'(k, m) = c || c; c = E(k, m) \\ D'(k, (c_0, c_1)) = D(k, c_0) \end{cases}$$
 – семантически стойкий шифр.

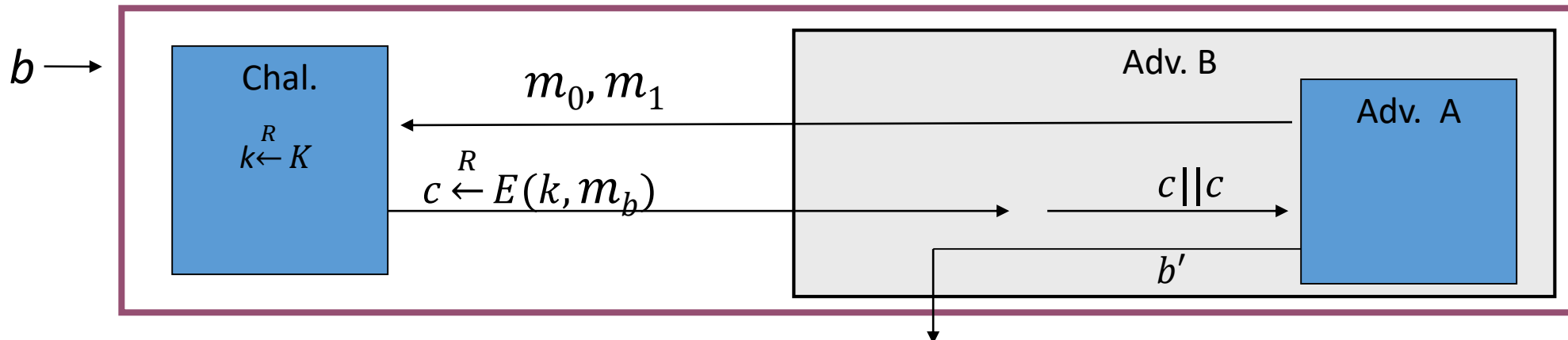
▷ От противного. Пусть E' – не семантически стойкий шифр. Тогда \exists противник A : $SSadv[A, E'] \geq e$, где e – не пренебрежимо малая величина.

Построим эффективный алгоритм B для игры против семантической стойкости шифра E с использованием алгоритма A , показав тем самым что E – не семантический стойкий \Rightarrow противоречие $\Rightarrow E'$ – семантический стойкий. ◁

Доказательства сведением (Reduction proof)

Пусть $E = (E, D)$ - вычислимый семантически стойкий шифр на (K, M, C) .
Тогда $E' = (E', D')$: $\begin{cases} (c_0, c_1) = E'(k, m) = c || c; c = E(k, m) \\ D'(k, (c_0, c_1)) = D(k, c_0) \end{cases}$ —
семантически стойкий шифр.

$\text{SSadv}[A, E'] \geq e$, где e – не пренебрежимо малая величина.



$$\text{SSadv}[B, E] = \text{SSadv}[A, E'] \geq e$$

Принципы доказательства стойкости и построения атак

- Строим атаку имея некоторые данные (примитив **A** нестойкий)
 - Просто строим атаку в указанной модели примитива **A**.

или

- Строим доказательство сведением (примитив **A** стойкий в некотором предположении стойкости другого примитива **B**; **стойкость B => стойкость A**).
 - От противного. Предполагаем, что примитив **A НЕ стойкий**.
 - Используя этот факт пытаемся построить **атаку** на примитив **B**.
 - Построили. Следовательно на **B** есть атака и он **НЕ стойкий**.
Следовательно предположение, что **A НЕ стойкий** было неверно, и **A – стойкий**.

Связь модели семантической стойкости с реальными атаками

- Есть ли связь между семантической стойкостью и, скажем, стойкостью к восстановлению сообщений (расшифровки ОТ по ШТ)
- Можно ли описать атаку на восстановление сообщений в терминах игры?

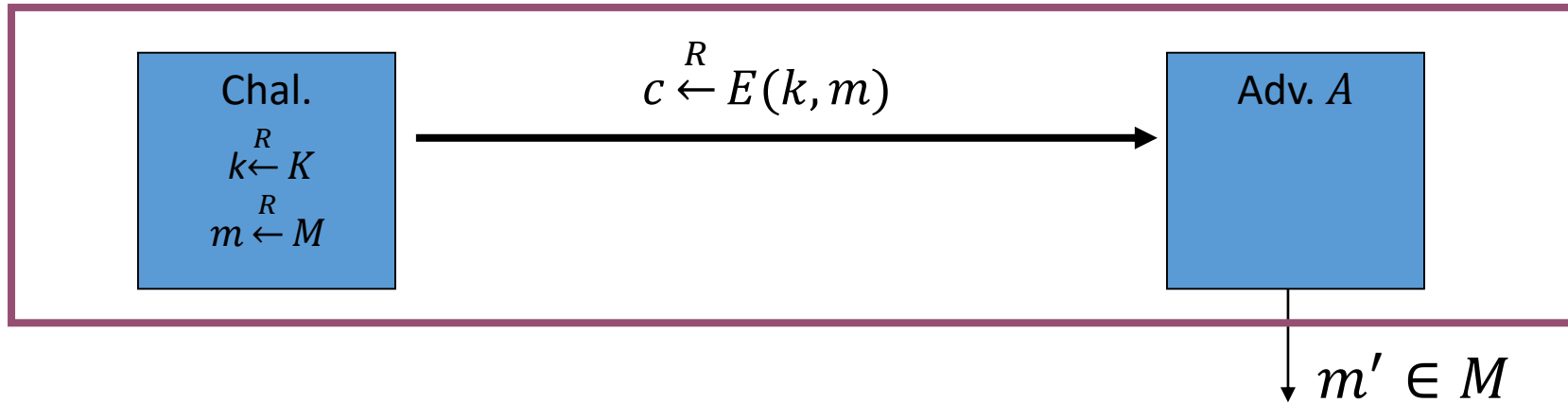
Восстановление сообщений

Атака на восстановление сообщений: имея зашифрованное сообщение $c \leftarrow E(k, t)$, $t \in M$, восстановить сообщение t , с вероятностью больше $1/|M|$.

Опишем игру на восстановление сообщений.

- Претендент вычисляет $t \xleftarrow{R} M$, $k \xleftarrow{R} K$, $c \xleftarrow{R} E(k, t)$ и отправляет c противнику.
- Противник возвращает t' как результат игры.

Восстановление сообщений

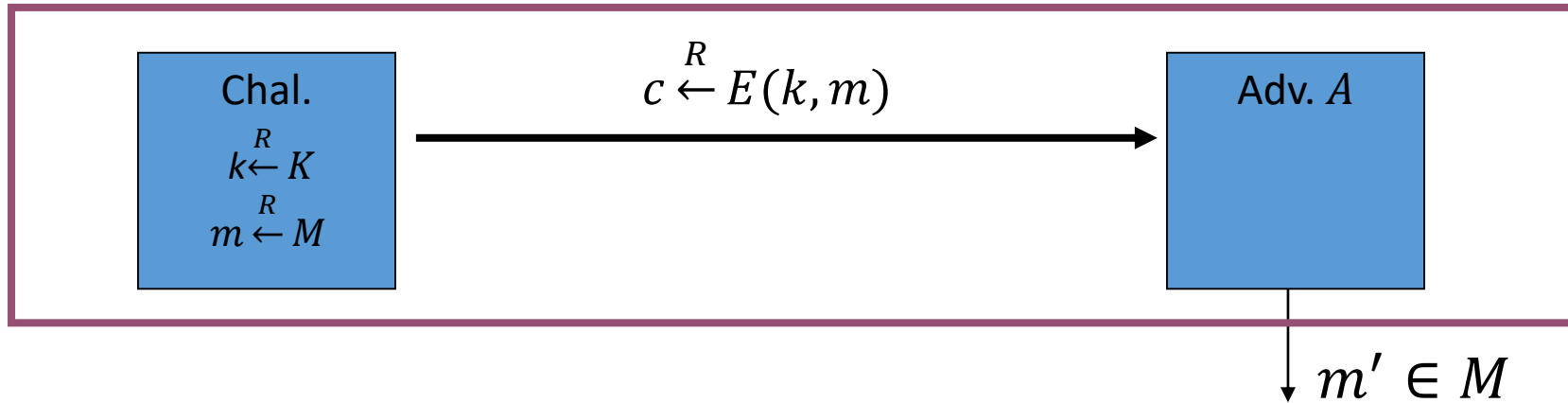


Пусть W – событие, при котором $m' = m$.

Преимуществом алгоритма A против шифра E при атаке на восстановление сообщений является величина

$$\text{MRadv}[A, E] = \left| \Pr[W] - \frac{1}{|M|} \right|$$

Восстановление сообщений



$$\text{MRadv}[A, E] = \left| \Pr[W] - \frac{1}{|M|} \right|$$

Шифр E называется **стойким к атаке на восстановление сообщений**, если $\forall A$ величина $\text{MRadv}[A, E] < \epsilon$, где ϵ - пренебрежимо малая величина.

Восстановление сообщений

Теорема 1.8. Если шифр $E = (E, D)$ семантически стойкий на (K, M, C) , то он стойкий к атаке на восстановление сообщений

▷ Покажем, что атака на восстановление сообщений даёт атаку на семантическую стойкость.

Пусть A – эффективный алгоритм. Обозначим p – вероятность выиграть игру на восстановление сообщений для алгоритма A :

$$\text{MRadv}[A, E] = \left| p - \frac{1}{|M|} \right|.$$

Построим эффективный алгоритм B для игры на семантическую стойкость против алгоритма E , для которого

$$\text{MRadv}[A, E] \leq \text{SSadv}[B, E].$$

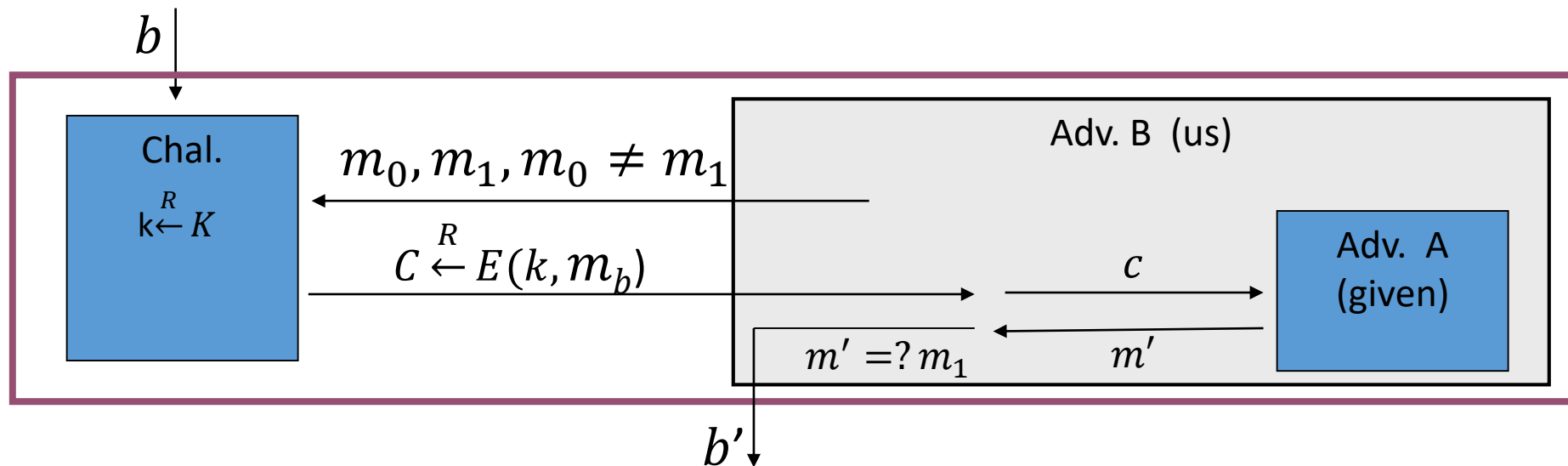
Восстановление сообщений

Теорема 1.8. Если шифр $E = (E, D)$ семантически стойкий на (K, M, C) , то он стойкий к атаке на восстановление сообщений

Построим алгоритм B . Алгоритм B генерирует два случайных сообщения m_0 и m_1 и отправляет их претенденту в игре на семантическую стойкость. Претендент отвечает шифртекстом c одного из сообщений, которых алгоритм B пересылает алгоритму A , получая восстановленное сообщение m' . Если $m' = m_0$ то выводит $b' = 0$, иначе $b' = 1$.

Восстановление сообщений

Теорема 1.8. Если шифр $E = (E, D)$ семантически стойкий на (K, M, C) , то он стойкий к атаке на восстановление сообщений



Восстановление сообщений

Теорема 1.8. Если шифр $E = (E, D)$ семантически стойкий на (K, M, C) , то он стойкий к атаке на восстановление сообщений

Для $b = 0, 1$ пусть p_b - вероятность того, что алгоритм B выдаст значение $b' = 1$, при шифровании сообщения m_b . Тогда $SSadv[B, E] = |p_0 - p_1|$. С другой стороны, если c есть зашифрование m_1 то вероятность $p_1 = p$ (Вероятность выиграть игру на восстановление для A). Если же c есть зашифрование m_0 , то p_0 не зависит от m_1 и $p_0 = \Pr[m' = m_1] = 1/|M|$. Следовательно

$$SSadv[B, E] = |p_1 - p_0| = \left| \frac{1}{|M|} - p \right| = MRadv[A, E]$$

⇒ атака на восстановление сообщений даёт атаку на семантическую стойкость. ◁

Восстановление битов сообщения

Пусть $E = (E, D)$ шифр на (K, M, C) . $M = \{0,1\}^L$. Пусть $par(m)$ – произвольный предикат, вычисляющий 1 бит информации об открытом тексте (Например функция вычисления бита чётности сообщения $m \in M$).

Определим игру на восстановление битов.

- Претендент вычисляет $m \stackrel{R}{\leftarrow} M, k \stackrel{R}{\leftarrow} K, c \stackrel{R}{\leftarrow} E(k, m)$ и отправляет c противнику.
- Противник возвращает $b' \in \{0,1\}$ как результат игры.

Пусть W – событие, при котором $b' = par(m)$.

Преимуществом алгоритма A против шифра E при атаке на восстановление битов является величина

$$\text{PARadv}[A, E] = |\Pr[W] - 1/2|$$

Восстановление битов сообщения

Пусть $E = (E, D)$ шифр на (K, M, C) . $M = \{0,1\}^L$. Пусть $par(m)$ – функция вычисления бита чётности сообщения $m \in M$ (здесь и далее $par: M \rightarrow \{0,1\}$: $par(m) = \bigoplus_{i=1}^L m_i$).

Шифр E называется **стойким к восстановлению битов**, если величина $PARadv[A, E] < \epsilon$, где ϵ – пренебрежимо малая величина.

Вычисление индивидуальных битов сообщений

Теорема 1.9. Если шифр $E = (E, D)$ семантически стойкий на (K, M, C) , то он стойкий к атаке на восстановление битов сообщения (Атака на восстановление битов сообщения даёт атаку на семантическую стойкость)

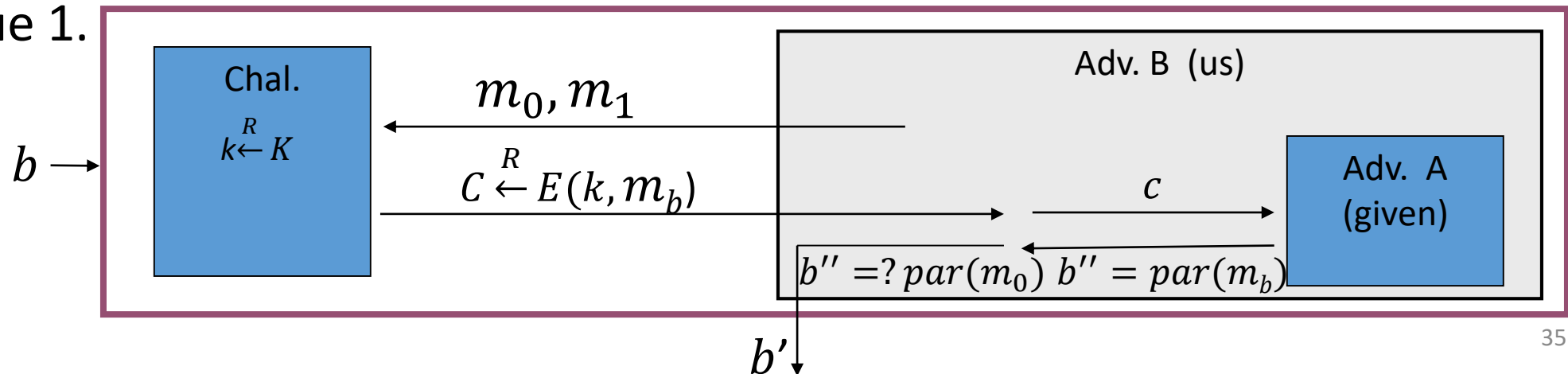
▷ От противного, имея алгоритм A в игре на восстановление битов построим эффективный алгоритм B для игры на семантическую стойкость против алгоритма E , для которого

$$\text{PARadv}[A, E] = \frac{1}{2} \text{SSadv}[B, E].$$

Вычисление индивидуальных битов сообщений

Теорема 1.9. Если шифр $E = (E, D)$ семантически стойкий на (K, M, C) , то он стойкий к атаке на восстановление битов сообщения (Атака на восстановление битов сообщения даёт атаку на семантическую стойкость)

Противник B генерирует сообщения $m_0, m_1 \leftarrow m_0 \oplus (0^{L-1}1)$ и отправляет претенденту, получая шифртекст c , который он передаёт алгоритму A . После получения значения b'' если $b'' = \text{par}(m_0)$ то $b' = 0$, иначе 1.



Вычисление индивидуальных битов сообщений

Теорема 1.9. Если шифр $E = (E, D)$ семантически стойкий на (K, M, C) , то он стойкий к атаке на восстановление битов сообщения (Атака на восстановление битов сообщения даёт атаку на семантическую стойкость)

Пусть A : $\text{PARadv}[A, E] = \epsilon$, т.е. вероятность угадать чётность есть $\frac{1}{2} + \epsilon$.

Для $b = 0, 1$ пусть p_b - вероятность того, что алгоритм B выдаст значение $b' = 1$. Тогда $\text{SSadv}[B, E] = |p_1 - p_0| = 2\epsilon = 2\text{PARadv}[A, E]$.

$$p_0 = \frac{1}{2} + \epsilon \text{ (угадали чётность } m_0),$$

$$p_1 = 1 - \frac{1}{2} + \epsilon = \frac{1}{2} - \epsilon \text{ (не угадали чётность } m_1).$$

⇒ атака на восстановление даёт атаку на семантическую стойкость. ◁

Семантическая стойкость (альтернативная формулировка)

Теорема 1.10. (обобщение 1.9) Пусть задана игра на семантическую стойкость для алгоритма A против шифра $E = (E, D)$ на (K, M, C) .

Определим $SSadv^*[A, E] = \left| \Pr[W] - \frac{1}{2} \right|$, где W — событие, при котором $b' = b$. Тогда $SSadv[A, E] = 2 * SSadv^*[A, E]$

▷ доказательство аналогично **Теореме 1.9.** ◁

Выводы

- Модель абсолютно стойкого шифра делает его сложно применимым в практическом смысле
 - Требуется размер ключа равный размеру сообщения
 - Невозможно добиться стойкости при переменной длине сообщений
- Семантически стойкий шифр – ослабленная модель абсолютно стойкого шифра, пригодная для практического применения
 - Стойкость к восстановлению сообщений
 - Стойкость к восстановлению битов сообщений
- Игровая модель – модель, позволяющая вводить определения стойкости для криптографических примитивов
 - Доказательства стойкости методом сведения (reduction)
 - Построение атак через моделирование игры

Тест.

Определение абсолютной стойкости через вероятность (без предиката):

Пусть $E = (E, D)$ – шифр
шеннона над
 (K, M, C) , $k \in_R K$

$\forall m_0, m_1 \in M$ и $c \in C$ имеем:

$$\Pr[???] = \Pr[????]$$

- Положить телефон экраном вниз справа от себя
- Не разговаривать с соседями
- Не пользоваться конспектами и электронными устройствами
- Написать номер (по таблице) и ФИО на листочке
- Написать краткий ответ на вопрос
- Дождаться окончания теста

Тест.

Определение абсолютной стойкости через предикат

- Положить телефон экраном вниз справа от себя
- Не разговаривать с соседями
- Не пользоваться конспектами и электронными устройствами
- Написать номер (по таблице) и ФИО на листочке
- Написать краткий ответ на вопрос
- Дождаться окончания теста