Лекция 9. Доказательства с нулевым разглашением (zero-knowledge proofs)

Косолапов Ю.В.

ЮФУ

12 ноября 2020 г.

Содержание

Для чего нужны Σ-протоколы

Для чего нужны Σ-протоколы

- Имеется доказывающая сторона P и **честная** проверяющая сторона V.

Для чего нужны Σ-протоколы

- Имеется доказывающая сторона P и **честная** проверяющая сторона V.
- Доказывающая сторона P должна убедить проверяющую сторону в **истинности некоторого утверждения**, не раскрывая какой-либо новой для V информации.

Для чего нужны Σ-протоколы

- Имеется доказывающая сторона P и **честная** проверяющая сторона V.
- Доказывающая сторона P должна убедить проверяющую сторону в **истинности некоторого утверждения**, не раскрывая какой-либо новой для V информации.

Example

Для чего нужны Σ-протоколы

- Имеется доказывающая сторона P и **честная** проверяющая сторона V.
- Доказывающая сторона P должна убедить проверяющую сторону в **истинности некоторого утверждения**, не раскрывая какой-либо новой для V информации.

Example

- В протоколе идентификации P доказывает, что знает секретный ключ. Здесь утверждение – «*Я знаю секретный ключ для этого публичного ключа*».

Для чего нужны Σ-протоколы

- Имеется доказывающая сторона P и **честная** проверяющая сторона V.
- Доказывающая сторона P должна убедить проверяющую сторону в **истинности некоторого утверждения**, не раскрывая какой-либо новой для V информации.

Example

- В протоколе идентификации P доказывает, что знает секретный ключ. Здесь утверждение – «*Я знаю секретный ключ для этого публичного ключа*».
- Но возможны и другие утверждения: «*Я знаю секретный ключ для это публичного ключа или секретный ключ для того секретного ключа, и ключи эти разные*».

Обозначение

Обозначение

- Пусть $R = \{(v, w)\} \subseteq \mathcal{V} \times \mathcal{W}$ — бинарное отношение,
  - $v \in \mathcal{V}$ — общее для V и P значение (например, публичный ключ)
  - $w \in \mathcal{W}$ — **свидетельство** (witness), которое известно только P (например, секретный ключ).

Обозначение

- Пусть $R = \{(v, w)\} \subseteq \mathcal{V} \times \mathcal{W}$ — бинарное отношение,
  - $v \in \mathcal{V}$ — общее для V и P значение (например, публичный ключ)
  - $w \in \mathcal{W}$ — **свидетельство** (witness), которое известно только P (например, секретный ключ).
- T.e. пара (v, w) принадлежит отношению R, если (нестрого) секретный ключ w соответствует публичному ключу v.

Обозначение

- Пусть $R = \{(v, w)\} \subseteq \mathcal{V} \times \mathcal{W}$ — бинарное отношение,
  - $v \in \mathcal{V}$ — общее для V и P значение (например, публичный ключ)
  - $w \in \mathcal{W}$ — **свидетельство** (witness), которое известно только P (например, секретный ключ).
- Т.е. пара (v, w) принадлежит отношению R, если (нестрого) секретный ключ w соответствует публичному ключу v.
- Определим язык

$$L_R = \{v \in \mathcal{V} : \exists_{w \in \mathcal{W}} (v, w) \in R\},$$

соответствующий отношению R.

Схема Σ-протокола

Предположения:

$$P : (v, w) \in R$$
$$V : (v \in \mathcal{V})$$

Схема Σ-протокола

Предположения:

$$P : (v, w) \in R$$
$$V : (v \in \mathcal{V})$$

Σ-протокол:

Схема Σ-протокола

Предположения:

$$P : (v, w) \in R$$
$$V : (v \in \mathcal{V})$$

Σ-протокол:

- $P : u_P \leftarrow \mathcal{U}$ (выбирается случайное значение, «краткосрочный ключ»)

Схема Σ-протокола

Предположения:

$$P : (v, w) \in R$$
$$V : (v \in \mathcal{V})$$

Σ-протокол:

- $P : u_P \leftarrow \mathcal{U}$ (выбирается случайное значение, «краткосрочный ключ»)
- $P : a \leftarrow \alpha(v, w, u_P)$ (вычисляется значение анонса)

Схема Σ-протокола

Предположения:

$$P : (v, w) \in R$$
$$V : (v \in \mathcal{V})$$

Σ-протокол:

- $P : u_P \leftarrow \mathcal{U}$ (выбирается случайное значение, «краткосрочный ключ»)
- $P : a \leftarrow \alpha(v, w, u_P)$ (вычисляется значение анонса)
- $P \rightarrow V : a$ (анонс)

Схема Σ-протокола

Предположения:

$$P : (v, w) \in R$$
$$V : (v \in \mathcal{V})$$

Σ-протокол:

- $P : u_P \leftarrow \mathcal{U}$ (выбирается случайное значение, «краткосрочный ключ»)
- $P : a \leftarrow \alpha(v, w, u_P)$ (вычисляется значение анонса)
- $P \rightarrow V : a$ (анонс)
- $V : c \in_R C$ (случайно выбирается значение запроса)

Схема Σ-протокола

Предположения:

$$P : (v, w) \in R$$
$$V : (v \in \mathcal{V})$$

Σ-протокол:

- $P : u_P \leftarrow \mathcal{U}$ (выбирается случайное значение, «краткосрочный ключ»)
- $P : a \leftarrow \alpha(v, w, u_P)$ (вычисляется значение анонса)
- $P \rightarrow V : a$ (анонс)
- $V : c \in_R C$ (случайно выбирается значение запроса)
- $V \rightarrow P : c$ (запрос)

Схема Σ-протокола

Предположения:

$$P : (v, w) \in R$$
$$V : (v \in \mathcal{V})$$

Σ-протокол:

- $P : u_P \leftarrow \mathcal{U}$ (выбирается случайное значение, «краткосрочный ключ»)
- $P : a \leftarrow \alpha(v, w, u_P)$ (вычисляется значение анонса)
- $P \rightarrow V : a$ (анонс)
- $V : c \in_R C$ (случайно выбирается значение запроса)
- $V \rightarrow P : c$ (запрос)
- $P : r \leftarrow \rho(v, w, c, u_P)$ (вычисляется значение ответа)

Схема Σ-протокола

Предположения:

$$P : (v, w) \in R$$
$$V : (v \in \mathcal{V})$$

Σ-протокол:

- $P : u_P \leftarrow \mathcal{U}$ (выбирается случайное значение, «краткосрочный ключ»)
- $P : a \leftarrow \alpha(v, w, u_P)$ (вычисляется значение анонса)
- $P \rightarrow V : a$ (анонс)
- $V : c \in_R C$ (случайно выбирается значение запроса)
- $V \rightarrow P : c$ (запрос)
- $P : r \leftarrow \rho(v, w, c, u_P)$ (вычисляется значение ответа)
- $P \rightarrow V : r$ (ответ)

Схема Σ-протокола

Предположения:

$$P : (v, w) \in R$$
$$V : (v \in \mathcal{V})$$

Σ-протокол:

- $P : u_P \leftarrow \mathcal{U}$ (выбирается случайное значение, «краткосрочный ключ»)
- $P : a \leftarrow \alpha(v, w, u_P)$ (вычисляется значение анонса)
- $P \rightarrow V : a$ (анонс)
- $V : c \in_R C$ (случайно выбирается значение запроса)
- $V \rightarrow P : c$ (запрос)
- $P : r \leftarrow \rho(v, w, c, u_P)$ (вычисляется значение ответа)
- $P \rightarrow V : r$ (ответ)
- $V : \phi(v, a, c, r)?$ (проверка истинности предиката ϕ)

Схема Σ-протокола

Предположения:

$$P : (v, w) \in R$$
$$V : (v \in \mathcal{V})$$

Σ-протокол:

- $P : u_P \leftarrow \mathcal{U}$ (выбирается случайное значение, «краткосрочный ключ»)
- $P : a \leftarrow \alpha(v, w, u_P)$ (вычисляется значение анонса)
- $P \rightarrow V : a$ (анонс)
- $V : c \in_R C$ (случайно выбирается значение запроса)
- $V \rightarrow P : c$ (запрос)
- $P : r \leftarrow \rho(v, w, c, u_P)$ (вычисляется значение ответа)
- $P \rightarrow V : r$ (ответ)
- $V : \phi(v, a, c, r)?$ (проверка истинности предиката ϕ)

Разговор (a, c, r) допустимый, если предикат ϕ истинен.

Определение

Определение Σ-протокола

Σ-протокол для отношения R — это протокол, изображенный на предыдущем слайде, для которого выполняются следующие свойства:

Определение

Определение Σ-протокола

Σ-протокол для отношения R — это протокол, изображенный на предыдущем слайде, для которого выполняются следующие свойства:

- **Полнота**. Если P и V следуют протоколу, то V всегда принимает разговор (т.е. разговор (a, c, r) всегда допустимый).

Определение

Определение Σ-протокола

Σ-протокол для отношения R — это протокол, изображенный на предыдущем слайде, для которого выполняются следующие свойства:

- **Полнота**. Если P и V следуют протоколу, то V всегда принимает разговор (т.е. разговор (a, c, r) всегда допустимый).
- **Корректность**. Существует полиномиальный алгоритм E (экстрактор, extractor), который для любого $v \in \mathcal{V}$ и двух любых допустимых разговоров (a, c, r) и (a, c', r') $(c \neq c')$ находит такое свидетельство w, что $(v, w) \in R$.

Определение

Определение Σ-протокола

Σ-протокол для отношения R — это протокол, изображенный на предыдущем слайде, для которого выполняются следующие свойства:

- **Полнота**. Если P и V следуют протоколу, то V всегда принимает разговор (т.е. разговор (a, c, r) всегда допустимый).
- **Корректность**. Существует полиномиальный алгоритм E (экстрактор, extractor), который для любого $v \in \mathcal{V}$ и двух любых допустимых разговоров (a, c, r) и (a, c', r') $(c \neq c')$ находит такое свидетельство w, что $(v, w) \in R$.
- **Особое ZK-свойство**. Существует полиномиальный алгоритм S (симулятор, simulator), который для любого $v \in L_R$ и любого $c \in C$ находит допустимый разговор (a, c, r), который имеет такую же вероятность появления при общении P (использует (v, w)) и V (использует v). Для $v \in \mathcal{V} \backslash L_R$ требуется, чтобы S мог построить какой-нибудь допустимый разговор для любого c .

Определение

Определение Σ-протокола

Σ-протокол для отношения R — это протокол, изображенный на предыдущем слайде, для которого выполняются следующие свойства:

- **Полнота**. Если P и V следуют протоколу, то V всегда принимает разговор (т.е. разговор (a, c, r) всегда допустимый).
- **Корректность**. Существует полиномиальный алгоритм E (экстрактор, extractor), который для любого $v \in \mathcal{V}$ и двух любых допустимых разговоров (a, c, r) и (a, c', r') $(c \neq c')$ находит такое свидетельство w, что $(v, w) \in R$.
- **Особое ZK-свойство**. Существует полиномиальный алгоритм S (симулятор, simulator), который для любого $v \in L_R$ и любого $c \in C$ находит допустимый разговор (a, c, r), который имеет такую же вероятность появления при общении P (использует (v, w)) и V (использует v). Для $v \in \mathcal{V} \backslash L_R$ требуется, чтобы S мог построить какой-нибудь допустимый разговор для любого c .

Если $|C| = 1$, то протокол называется **тривиальным**.

Пояснение свойства **Корректность**

Пояснение свойства **Корректность**

- **Типа аксиомы**: по v найти w сложно (по публичному ключу найти секретный сложно).

Пояснение свойства **Корректность**

- **Типа аксиомы**: по v найти w сложно (по публичному ключу найти секретный сложно).
- И пусть существует полиномиальный экстрактор E (он известен всем):

$$\big(v, (a, c, r), (a, c', r')\big) \to w.$$

Пояснение свойства **Корректность**

- **Типа аксиомы**: по v найти w сложно (по публичному ключу найти секретный сложно).
- И пусть существует полиномиальный экстрактор E (он известен всем):

$$\big(v, (a, c, r), (a, c', r')\big) \to w.$$

- Если P', не зная v, может подготовить два допустимых разговора (a, c, r) и (a, c', r'), то P' далее может воспользоваться экстрактором E для нахождения w.

Пояснение свойства **Корректность**

- **Типа аксиомы**: по v найти w сложно (по публичному ключу найти секретный сложно).
- И пусть существует полиномиальный экстрактор E (он известен всем):

$$\big(v, (a, c, r), (a, c', r')\big) \to w.$$

- Если P', не зная v, может подготовить два допустимых разговора (a, c, r) и (a, c', r'), то P' далее может воспользоваться экстрактором E для нахождения w.
- Но это противоречит «типа аксиоме».

Пояснение свойства **Корректность**

- **Типа аксиомы**: по v найти w сложно (по публичному ключу найти секретный сложно).
- И пусть существует полиномиальный экстрактор E (он известен всем):

$$\big(v, (a, c, r), (a, c', r')\big) \to w.$$

- Если P', не зная v, может подготовить два допустимых разговора (a, c, r) и (a, c', r'), то P' далее может воспользоваться экстрактором E для нахождения w.
- Но это противоречит «типа аксиоме».

Вывод

Поэтому, при выполнении «типа аксиомы» и наличии полиномиального E, говорят, что без знания секретного ключа можно доказать истинность утверждения с вероятностью не более $1/n$, где $n = |C|$.

Пояснение особенности ZK-свойства (**Особое ZK-свойство**)

Пояснение особенности ZK-свойства (**Особое ZK-свойство**)

- Для **простого ZK-свойства** симулятору S на вход подается **только публичный ключ** $v \in \mathcal{V}$, а далее симулятор сам выбирает запрос $c \in C$ и строит допустимый разговор (a, c, r).

Пояснение особенности ZK-свойства (**Особое ZK-свойство**)

- Для **простого ZK-свойства** симулятору S на вход подается **только публичный ключ** $v \in \mathcal{V}$, а далее симулятор сам выбирает запрос $c \in C$ и строит допустимый разговор (a, c, r).
- Для **особого ZK-свойства** симулятору S подается **пара публичный ключ-запрос** $(v, c) \in \mathcal{V} \times C$, на основе которой он строит допустимый разговор (a, c, r). Это свойство считается более строгим.

Пояснение особенности ZK-свойства (**Особое ZK-свойство**)

- Для **простого ZK-свойства** симулятору S на вход подается **только публичный ключ** $v \in \mathcal{V}$, а далее симулятор сам выбирает запрос $c \in C$ и строит допустимый разговор (a, c, r).
- Для **особого ZK-свойства** симулятору S подается **пара публичный ключ-запрос** $(v, c) \in \mathcal{V} \times C$, на основе которой он строит допустимый разговор (a, c, r). Это свойство считается более строгим.

Вывод

Можно ли преобразовать протокол с ZK-свойством к протоколу с особым ZK-свойством?

Пояснение особенности ZK-свойства (**Особое ZK-свойство**)

- Для **простого ZK-свойства** симулятору S на вход подается **только публичный ключ** $v \in \mathcal{V}$, а далее симулятор сам выбирает запрос $c \in C$ и строит допустимый разговор (a, c, r).
- Для **особого ZK-свойства** симулятору S подается **пара публичный ключ-запрос** $(v, c) \in \mathcal{V} \times C$, на основе которой он строит допустимый разговор (a, c, r). Это свойство считается более строгим.

Вывод

Можно ли преобразовать протокол с ZK-свойством к протоколу с особым ZK-свойством? Да (см. далее).

Схема **преобразованного** Σ-протокола

Предположения:

$$P : (v, w) \in R$$

$$V : (v \in \mathcal{V})$$

C — аддитивная группа

Схема **преобразованного** Σ-протокола

Предположения:

$$P : (v, w) \in R$$
$$V : (v \in \mathcal{V})$$
$$C - \text{аддитивная группа}$$

Σ-протокол:

Схема **преобразованного** Σ-протокола

Предположения:

$$P : (v, w) \in R$$
$$V : (v \in \mathcal{V})$$
$$C - \text{аддитивная группа}$$

Σ-протокол:

- $P : u_P \leftarrow \mathcal{U}$ (выбирается случайное значение, «краткосрочный ключ»)

Схема **преобразованного** Σ-протокола

Предположения:

$$P : (v, w) \in R$$
$$V : (v \in \mathcal{V})$$
$$C - \text{аддитивная группа}$$

Σ-протокол:

- $P : u_P \leftarrow \mathcal{U}$ (выбирается случайное значение, «краткосрочный ключ»)
- $P : a \leftarrow \alpha(v, w, u_P), c_P \in_R C$ (вычисляется значение анонса)

Схема **преобразованного** Σ-протокола

Предположения:

$$P : (v, w) \in R$$
$$V : (v \in \mathcal{V})$$
$$C - \text{аддитивная группа}$$

Σ-протокол:

- $P : u_P \leftarrow \mathcal{U}$ (выбирается случайное значение, «краткосрочный ключ»)
- $P : a \leftarrow \alpha(v, w, u_P), c_P \in_R C$ (вычисляется значение анонса)
- $P \rightarrow V : a, c_P$ (анонс)

Схема **преобразованного** Σ-протокола

Предположения:

$$P : (v, w) \in R$$
$$V : (v \in \mathcal{V})$$
$$C - \text{аддитивная группа}$$

Σ-протокол:

- $P : u_P \leftarrow \mathcal{U}$ (выбирается случайное значение, «краткосрочный ключ»)
- $P : a \leftarrow \alpha(v, w, u_P), c_P \in_R C$ (вычисляется значение анонса)
- $P \rightarrow V : a, c_P$ (анонс)
- $V : c_V \in_R C$ (случайно выбирается значение запроса)

Схема **преобразованного** Σ-протокола

Предположения:

$$P : (v, w) \in R$$
$$V : (v \in \mathcal{V})$$
$$C - \text{аддитивная группа}$$

Σ-протокол:

- $P : u_P \leftarrow \mathcal{U}$ (выбирается случайное значение, «краткосрочный ключ»)
- $P : a \leftarrow \alpha(v, w, u_P), c_P \in_R C$ (вычисляется значение анонса)
- $P \rightarrow V : a, c_P$ (анонс)
- $V : c_V \in_R C$ (случайно выбирается значение запроса)
- $V \rightarrow P : c_V$ (запрос)

Схема **преобразованного** Σ-протокола

Предположения:

$$P : (v, w) \in R$$
$$V : (v \in \mathcal{V})$$
$$C - \text{аддитивная группа}$$

Σ-протокол:

- $P : u_P \leftarrow \mathcal{U}$ (выбирается случайное значение, «краткосрочный ключ»)
- $P : a \leftarrow \alpha(v, w, u_P), c_P \in_R C$ (вычисляется значение анонса)
- $P \rightarrow V : a, c_P$ (анонс)
- $V : c_V \in_R C$ (случайно выбирается значение запроса)
- $V \rightarrow P : c_V$ (запрос)
- $P : r \leftarrow \rho(v, w, c_P + c_V, u_P)$ (вычисляется значение ответа)

Схема **преобразованного** Σ-протокола

Предположения:

$$P : (v, w) \in R$$
$$V : (v \in \mathcal{V})$$
$$C - \text{аддитивная группа}$$

Σ-протокол:

- $P : u_P \leftarrow \mathcal{U}$ (выбирается случайное значение, «краткосрочный ключ»)
- $P : a \leftarrow \alpha(v, w, u_P), c_P \in_R C$ (вычисляется значение анонса)
- $P \rightarrow V : a, c_P$ (анонс)
- $V : c_V \in_R C$ (случайно выбирается значение запроса)
- $V \rightarrow P : c_V$ (запрос)
- $P : r \leftarrow \rho(v, w, c_P + c_V, u_P)$ (вычисляется значение ответа)
- $P \rightarrow V : r$ (ответ)

Схема **преобразованного** Σ-протокола

Предположения:

$$P : (v, w) \in R$$
$$V : (v \in \mathcal{V})$$
$$C - \text{аддитивная группа}$$

Σ-протокол:

- $P : u_P \leftarrow \mathcal{U}$ (выбирается случайное значение, «краткосрочный ключ»)
- $P : a \leftarrow \alpha(v, w, u_P), c_P \in_R C$ (вычисляется значение анонса)
- $P \rightarrow V : a, c_P$ (анонс)
- $V : c_V \in_R C$ (случайно выбирается значение запроса)
- $V \rightarrow P : c_V$ (запрос)
- $P : r \leftarrow \rho(v, w, c_P + c_V, u_P)$ (вычисляется значение ответа)
- $P \rightarrow V : r$ (ответ)
- $V : \phi(v, a, c_P + c_V, r)?$ (проверка истинности предиката ϕ)

Утверждение

Трансформированный протокол является Σ-протоколом, если оригинальный протокол обладает свойствами полноты, корректности и ZK-свойством.

Доказательство:

Утверждение

Трансформированный протокол является Σ-протоколом, если оригинальный протокол обладает свойствами полноты, корректности и ZK-свойством.

Доказательство:

- **Полнота** протокола очевидна: просто значение c заменено на $c_P + c_V$.

Утверждение

Трансформированный протокол является Σ-протоколом, если оригинальный протокол обладает свойствами полноты, корректности и ZK-свойством.

Доказательство:

- **Полнота** протокола очевидна: просто значение c заменено на $c_P + c_V$.
- **Корректность**: пусть (a, c_P, c_V, r) и (a, c_P, c'_V, r') — допустимые разговоры. Определим $c = c_P + c_V$ и $c' = c_P + c'_V$. Так как $c \neq c'$, то корректность оригинального протокола гарантирует нахождение свидетельства w по корректным разговорам (a, c, r) и (a, c', r').

Утверждение

Трансформированный протокол является Σ-протоколом, если оригинальный протокол обладает свойствами полноты, корректности и ZK-свойством.

Доказательство:

- **Полнота** протокола очевидна: просто значение c заменено на $c_P + c_V$.
- **Корректность**: пусть (a, c_P, c_V, r) и (a, c_P, c'_V, r') — допустимые разговоры. Определим $c = c_P + c_V$ и $c' = c_P + c'_V$. Так как $c \neq c'$, то корректность оригинального протокола гарантирует нахождение свидетельства w по корректным разговорам (a, c, r) и (a, c', r').
- **Особое ZK-свойство**. Построим симулятор S' для трансформированного протокола по симулятору S для оригинального протокола:

Утверждение

Трансформированный протокол является Σ-протоколом, если оригинальный протокол обладает свойствами полноты, корректности и ZK-свойством.

Доказательство:

- **Полнота** протокола очевидна: просто значение c заменено на $c_P + c_V$.
- **Корректность**: пусть (a, c_P, c_V, r) и (a, c_P, c'_V, r') — допустимые разговоры. Определим $c = c_P + c_V$ и $c' = c_P + c'_V$. Так как $c \neq c'$, то корректность оригинального протокола гарантирует нахождение свидетельства w по корректным разговорам (a, c, r) и (a, c', r').
- **Особое ZK-свойство**. Построим симулятор S' для трансформированного протокола по симулятору S для оригинального протокола:
  - S' получает на вход **заданный** запрос c_V;

Утверждение

Трансформированный протокол является Σ-протоколом, если оригинальный протокол обладает свойствами полноты, корректности и ZK-свойством.

Доказательство:

- **Полнота** протокола очевидна: просто значение c заменено на $c_P + c_V$.
- **Корректность**: пусть (a, c_P, c_V, r) и (a, c_P, c'_V, r') — допустимые разговоры. Определим $c = c_P + c_V$ и $c' = c_P + c'_V$. Так как $c \neq c'$, то корректность оригинального протокола гарантирует нахождение свидетельства w по корректным разговорам (a, c, r) и (a, c', r').
- **Особое ZK-свойство**. Построим симулятор S' для трансформированного протокола по симулятору S для оригинального протокола:
  - S' получает на вход **заданный** запрос c_V;
  - S' вызывает S, который генерирует допустимый разговор (a, c, r) (S выбирает c сам случайным образом);

Утверждение

Трансформированный протокол является Σ-протоколом, если оригинальный протокол обладает свойствами полноты, корректности и ZK-свойством.

Доказательство:

- **Полнота** протокола очевидна: просто значение c заменено на $c_P + c_V$.
- **Корректность**: пусть (a, c_P, c_V, r) и (a, c_P, c'_V, r') — допустимые разговоры. Определим $c = c_P + c_V$ и $c' = c_P + c'_V$. Так как $c \neq c'$, то корректность оригинального протокола гарантирует нахождение свидетельства w по корректным разговорам (a, c, r) и (a, c', r').
- **Особое ZK-свойство**. Построим симулятор S' для трансформированного протокола по симулятору S для оригинального протокола:
  - S' получает на вход **заданный** запрос c_V;
  - S' вызывает S, который генерирует допустимый разговор (a, c, r) (S выбирает c сам случайным образом);
  - S' вычисляет: $c_P = c - c_V$;

Утверждение

Трансформированный протокол является Σ-протоколом, если оригинальный протокол обладает свойствами полноты, корректности и ZK-свойством.

Доказательство:

- **Полнота** протокола очевидна: просто значение c заменено на $c_P + c_V$.
- **Корректность**: пусть (a, c_P, c_V, r) и (a, c_P, c'_V, r') — допустимые разговоры. Определим $c = c_P + c_V$ и $c' = c_P + c'_V$. Так как $c \neq c'$, то корректность оригинального протокола гарантирует нахождение свидетельства w по корректным разговорам (a, c, r) и (a, c', r').
- **Особое ZK-свойство**. Построим симулятор S' для трансформированного протокола по симулятору S для оригинального протокола:
  - S' получает на вход **заданный** запрос c_V;
  - S' вызывает S, который генерирует допустимый разговор (a, c, r) (S выбирает c сам случайным образом);
  - S' вычисляет: $c_P = c - c_V$;
  - (a, c_P, c_V, r) — допустимый разговор;

Утверждение

Трансформированный протокол является Σ-протоколом, если оригинальный протокол обладает свойствами полноты, корректности и ZK-свойством.

Доказательство:

- **Полнота** протокола очевидна: просто значение c заменено на $c_P + c_V$.
- **Корректность**: пусть (a, c_P, c_V, r) и (a, c_P, c'_V, r') — допустимые разговоры. Определим $c = c_P + c_V$ и $c' = c_P + c'_V$. Так как $c \neq c'$, то корректность оригинального протокола гарантирует нахождение свидетельства w по корректным разговорам (a, c, r) и (a, c', r').
- **Особое ZK-свойство**. Построим симулятор S' для трансформированного протокола по симулятору S для оригинального протокола:
  - S' получает на вход **заданный** запрос c_V;
  - S' вызывает S, который генерирует допустимый разговор (a, c, r) (S выбирает c сам случайным образом);
  - S' вычисляет: $c_P = c - c_V$;
  - (a, c_P, c_V, r) — допустимый разговор;
  - c_P принимает значение случайно и равновероятно из C, поэтому разговор имеет такое же распределение, как и разговор с реальным P.

Домашнее задание

Вариант протокола Шнорра с одной итерацией

- Пусть $\langle g \rangle$ — циклическая группа с порождающим элементом g, $n = |\langle g \rangle|$ – порядок этой группы (большое, желательно простое число). **Ключи доказывающей стороны**:
  - $sk_P = x$, $x \in_R \mathbb{Z}_n$ — случайно выбранный из группы элемент,
  - $pk_P = g^x = h$. Этот ключ передается проверяющей стороне в фазе регистрации.
- 1. $P : u \in_R \mathbb{Z}_n$, $a = g^u$
  2. $P \to V : a$
  3. $V : c \in_R \mathbb{Z}_n$
  4. $V \to P : c$
  5. $P : r = x + u \cdot c$
  6. $P \to V : r$
  7. $V : g^r \stackrel{?}{=} a^c \cdot h$

Домашнее задание

Вариант протокола Шнорра с одной итерацией

- Пусть $\langle g \rangle$ — циклическая группа с порождающим элементом g, $n = |\langle g \rangle|$ – порядок этой группы (большое, желательно простое число). **Ключи доказывающей стороны**:
  - $sk_P = x$, $x \in_R \mathbb{Z}_n$ — случайно выбранный из группы элемент,
  - $pk_P = g^x = h$. Этот ключ передается проверяющей стороне в фазе регистрации.
- 1. $P : u \in_R \mathbb{Z}_n$, $a = g^u$
  2. $P \to V : a$
  3. $V : c \in_R \mathbb{Z}_n$
  4. $V \to P : c$
  5. $P : r = x + u \cdot c$
  6. $P \to V : r$
  7. $V : g^r \stackrel{?}{=} a^c \cdot h$
- Этот протокол обладает свойствами полноты, корректности и ZK-свойством в случае честного V.

Домашнее задание

Вариант протокола Шнорра с одной итерацией

- Пусть $\langle g \rangle$ — циклическая группа с порождающим элементом g, $n = |\langle g \rangle|$ – порядок этой группы (большое, желательно простое число). **Ключи доказывающей стороны**:
  - $sk_P = x$, $x \in_R \mathbb{Z}_n$ — случайно выбранный из группы элемент,
  - $pk_P = g^x = h$. Этот ключ передается проверяющей стороне в фазе регистрации.
- 1. $P : u \in_R \mathbb{Z}_n$, $a = g^u$
  2. $P \to V : a$
  3. $V : c \in_R \mathbb{Z}_n$
  4. $V \to P : c$
  5. $P : r = x + u \cdot c$
  6. $P \to V : r$
  7. $V : g^r \stackrel{?}{=} a^c \cdot h$
- Этот протокол обладает свойствами полноты, корректности и ZK-свойством в случае честного V.
- В случае нечестного V этот протокол обладает свойствами полноты и корректности, но не обладает ZK-свойством (почему?).

Параллельная композиция Σ-протоколов

- Пусть $\langle g \rangle$ — циклическая группа с порождающим элементом g, $n = |\langle g \rangle|$ – порядок этой группы. **Ключи доказывающей стороны**:
  - $sk_P = x$, $x \in_R \mathbb{Z}_n$,
  - $pk_P = g^x = h$.

Параллельная композиция Σ-протоколов

- Пусть $\langle g \rangle$ — циклическая группа с порождающим элементом g, $n = |\langle g \rangle|$ – порядок этой группы. **Ключи доказывающей стороны**:
  - $sk_P = x$, $x \in_R \mathbb{Z}_n$,
  - $pk_P = g^x = h$.
- 1. $P : u_1 \in_R \mathbb{Z}_n, u_2 \in_R \mathbb{Z}_n$, $a_1 = g^{u_1}$, $a_2 = g^{u_2}$

Параллельная композиция Σ-протоколов

- Пусть $\langle g \rangle$ — циклическая группа с порождающим элементом g, $n = |\langle g \rangle|$ – порядок этой группы. **Ключи доказывающей стороны**:
  - $sk_P = x$, $x \in_R \mathbb{Z}_n$,
  - $pk_P = g^x = h$.
- 1. $P : u_1 \in_R \mathbb{Z}_n, u_2 \in_R \mathbb{Z}_n,\ a_1 = g^{u_1},\ a_2 = g^{u_2}$
  2. $P \to V : a_1, a_2$

Параллельная композиция Σ-протоколов

- Пусть $\langle g \rangle$ — циклическая группа с порождающим элементом g, $n = |\langle g \rangle|$ – порядок этой группы. **Ключи доказывающей стороны**:
  - $sk_P = x$, $x \in_R \mathbb{Z}_n$,
  - $pk_P = g^x = h$.
- 1. $P : u_1 \in_R \mathbb{Z}_n, u_2 \in_R \mathbb{Z}_n,\ a_1 = g^{u_1},\ a_2 = g^{u_2}$
  2. $P \to V : a_1, a_2$
  3. $V : c_1, c_2 \in_R \mathbb{Z}_n$

Параллельная композиция Σ-протоколов

- Пусть $\langle g \rangle$ — циклическая группа с порождающим элементом g, $n = |\langle g \rangle|$ – порядок этой группы. **Ключи доказывающей стороны**:
  - $sk_P = x$, $x \in_R \mathbb{Z}_n$,
  - $pk_P = g^x = h$.
- 1. $P : u_1 \in_R \mathbb{Z}_n, u_2 \in_R \mathbb{Z}_n,\ a_1 = g^{u_1},\ a_2 = g^{u_2}$
  2. $P \to V : a_1, a_2$
  3. $V : c_1, c_2 \in_R \mathbb{Z}_n$
  4. $V \to P : c_1, c_2$

Параллельная композиция Σ-протоколов

- Пусть $\langle g \rangle$ — циклическая группа с порождающим элементом g, $n = |\langle g \rangle|$ – порядок этой группы. **Ключи доказывающей стороны**:
  - $sk_P = x,\ x \in_R \mathbb{Z}_n,$
  - $pk_P = g^x = h.$
- 1. $P : u_1 \in_R \mathbb{Z}_n, u_2 \in_R \mathbb{Z}_n,\ a_1 = g^{u_1},\ a_2 = g^{u_2}$
  2. $P \to V : a_1, a_2$
  3. $V : c_1, c_2 \in_R \mathbb{Z}_n$
  4. $V \to P : c_1, c_2$
  5. $P : r_1 = u_1 + x \cdot c_1, r_2 = u_2 + x \cdot c_2$

Параллельная композиция Σ-протоколов

- Пусть $\langle g \rangle$ — циклическая группа с порождающим элементом g, $n = |\langle g \rangle|$ – порядок этой группы. **Ключи доказывающей стороны**:
  - $sk_P = x$, $x \in_R \mathbb{Z}_n$,
  - $pk_P = g^x = h$.
- 1. $P : u_1 \in_R \mathbb{Z}_n, u_2 \in_R \mathbb{Z}_n,\ a_1 = g^{u_1},\ a_2 = g^{u_2}$
  2. $P \to V : a_1, a_2$
  3. $V : c_1, c_2 \in_R \mathbb{Z}_n$
  4. $V \to P : c_1, c_2$
  5. $P : r_1 = u_1 + x \cdot c_1, r_2 = u_2 + x \cdot c_2$
  6. $P \to V : r_1, r_2$

Параллельная композиция Σ-протоколов

- Пусть $\langle g \rangle$ — циклическая группа с порождающим элементом g, $n = |\langle g \rangle|$ – порядок этой группы. **Ключи доказывающей стороны**:
  - $sk_P = x,\ x \in_R \mathbb{Z}_n,$
  - $pk_P = g^x = h.$
- 1. $P : u_1 \in_R \mathbb{Z}_n, u_2 \in_R \mathbb{Z}_n,\ a_1 = g^{u_1},\ a_2 = g^{u_2}$
  2. $P \to V : a_1, a_2$
  3. $V : c_1, c_2 \in_R \mathbb{Z}_n$
  4. $V \to P : c_1, c_2$
  5. $P : r_1 = u_1 + x \cdot c_1, r_2 = u_2 + x \cdot c_2$
  6. $P \to V : r_1, r_2$
  7. $V : g^{r_1} \stackrel{?}{=} a \cdot h^{c_1}, g^{r_2} \stackrel{?}{=} a \cdot h^{c_2}$

Параллельная композиция Σ-протоколов. Утверждение

Утверждение

Параллельная композиция на предыдущем слайде является Σ-протоколом для отношения $R = \{(h, x) : g^x = h\}$.

Доказательство (схема):

Параллельная композиция Σ-протоколов. Утверждение

Утверждение

Параллельная композиция на предыдущем слайде является Σ-протоколом для отношения $R = \{(h, x) : g^x = h\}$.

Доказательство (схема):

- **Полнота** следует из полноты протокола Шнорра.

Параллельная композиция Σ-протоколов. Утверждение

Утверждение

Параллельная композиция на предыдущем слайде является Σ-протоколом для отношения $R = \{(h, x) : g^x = h\}$.

Доказательство (схема):

- **Полнота** следует из полноты протокола Шнорра.
- **Корректность**. Пусть $(a_1, a_2, c_1, c, r_1, r_2)$ и $(a_1, a_2, c_1', c', r_1', r_2')$ допустимые разговоры для $(c_1, c_2) \neq (c_1', c_2')$. Тогда или $c_1 \neq c_1'$, или $c_2 \neq c_2'$. Для каждого отдельного случая можно повторить рассуждения для протокола Шнорра и получим, что можно найти логарифм от h (полиномиальный экстрактор E).

Параллельная композиция Σ-протоколов. Утверждение

Утверждение

Параллельная композиция на предыдущем слайде является Σ-протоколом для отношения $R = \{(h, x) : g^x = h\}$.

Доказательство (схема):

- **Полнота** следует из полноты протокола Шнорра.
- **Корректность**. Пусть $(a_1, a_2, c_1, c, r_1, r_2)$ и $(a_1, a_2, c_1', c', r_1', r_2')$ допустимые разговоры для $(c_1, c_2) \neq (c_1', c_2')$. Тогда или $c_1 \neq c_1'$, или $c_2 \neq c_2'$. Для каждого отдельного случая можно повторить рассуждения для протокола Шнорра и получим, что можно найти логарифм от h (полиномиальный экстрактор E).
- **Особое ZK-свойство**. Следует из того, что, по сути, рассматриваемый протокол — это два независимых выполнения протокола Шнорра. Каждый разговор имеет вероятность $\frac{1}{n^2}$.

AND-композиция Σ-протоколов

- Пусть $\langle g \rangle$ — циклическая группа с порождающим элементом g, $n = |\langle g \rangle|$ – порядок этой группы. **Ключи доказывающей стороны**:
  - $sk_P = (x_1, x_2)$, $x_1, x_2 \in_R \mathbb{Z}_n$,
  - $pk_P = (g^{x_1} = h_1, g^{x_2} = h_2)$.

AND-композиция Σ-протоколов

- Пусть $\langle g \rangle$ — циклическая группа с порождающим элементом g, $n = |\langle g \rangle|$ – порядок этой группы. **Ключи доказывающей стороны**:
  - $sk_P = (x_1, x_2)$, $x_1, x_2 \in_R \mathbb{Z}_n$,
  - $pk_P = (g^{x_1} = h_1, g^{x_2} = h_2)$.
- 1. $P : u_1 \in_R \mathbb{Z}_n, u_2 \in_R \mathbb{Z}_n$, $a_1 = g^{u_1}$, $a_2 = g^{u_2}$

AND-композиция Σ-протоколов

- Пусть $\langle g \rangle$ — циклическая группа с порождающим элементом g, $n = |\langle g \rangle|$ – порядок этой группы. **Ключи доказывающей стороны**:
  - $sk_P = (x_1, x_2)$, $x_1, x_2 \in_R \mathbb{Z}_n$,
  - $pk_P = (g^{x_1} = h_1, g^{x_2} = h_2)$.
- 1. $P : u_1 \in_R \mathbb{Z}_n, u_2 \in_R \mathbb{Z}_n$, $a_1 = g^{u_1}$, $a_2 = g^{u_2}$
  2. $P \to V : a_1, a_2$

AND-композиция Σ-протоколов

- Пусть $\langle g \rangle$ — циклическая группа с порождающим элементом g, $n = |\langle g \rangle|$ – порядок этой группы. **Ключи доказывающей стороны**:
  - $sk_P = (x_1, x_2)$, $x_1, x_2 \in_R \mathbb{Z}_n$,
  - $pk_P = (g^{x_1} = h_1, g^{x_2} = h_2)$.
- 1. $P : u_1 \in_R \mathbb{Z}_n, u_2 \in_R \mathbb{Z}_n$, $a_1 = g^{u_1}$, $a_2 = g^{u_2}$
  2. $P \to V : a_1, a_2$
  3. $V : c, \in_R \mathbb{Z}_n$

AND-композиция Σ-протоколов

- Пусть $\langle g \rangle$ — циклическая группа с порождающим элементом g, $n = |\langle g \rangle|$ – порядок этой группы. **Ключи доказывающей стороны**:
  - $sk_P = (x_1, x_2)$, $x_1, x_2 \in_R \mathbb{Z}_n$,
  - $pk_P = (g^{x_1} = h_1, g^{x_2} = h_2)$.
- 1. $P : u_1 \in_R \mathbb{Z}_n, u_2 \in_R \mathbb{Z}_n$, $a_1 = g^{u_1}$, $a_2 = g^{u_2}$
  2. $P \to V : a_1, a_2$
  3. $V : c, \in_R \mathbb{Z}_n$
  4. $V \to P : c$

AND-композиция Σ-протоколов

- Пусть $\langle g \rangle$ — циклическая группа с порождающим элементом g, $n = |\langle g \rangle|$ – порядок этой группы. **Ключи доказывающей стороны**:
  - $sk_P = (x_1, x_2)$, $x_1, x_2 \in_R \mathbb{Z}_n$,
  - $pk_P = (g^{x_1} = h_1, g^{x_2} = h_2)$.
- 1. $P : u_1 \in_R \mathbb{Z}_n, u_2 \in_R \mathbb{Z}_n$, $a_1 = g^{u_1}$, $a_2 = g^{u_2}$
  2. $P \to V : a_1, a_2$
  3. $V : c, \in_R \mathbb{Z}_n$
  4. $V \to P : c$
  5. $P : r_1 = u_1 + x_1 \cdot c, r_2 = u_2 + x_2 \cdot c$

AND-композиция Σ-протоколов

- Пусть $\langle g \rangle$ — циклическая группа с порождающим элементом g, $n = |\langle g \rangle|$ – порядок этой группы. **Ключи доказывающей стороны**:
  - $sk_P = (x_1, x_2)$, $x_1, x_2 \in_R \mathbb{Z}_n$,
  - $pk_P = (g^{x_1} = h_1, g^{x_2} = h_2)$.
- 1. $P : u_1 \in_R \mathbb{Z}_n, u_2 \in_R \mathbb{Z}_n$, $a_1 = g^{u_1}$, $a_2 = g^{u_2}$
  2. $P \to V : a_1, a_2$
  3. $V : c, \in_R \mathbb{Z}_n$
  4. $V \to P : c$
  5. $P : r_1 = u_1 + x_1 \cdot c, r_2 = u_2 + x_2 \cdot c$
  6. $P \to V : r_1, r_2$

AND-композиция Σ-протоколов

- Пусть $\langle g \rangle$ — циклическая группа с порождающим элементом g, $n = |\langle g \rangle|$ – порядок этой группы. **Ключи доказывающей стороны**:
  - $sk_P = (x_1, x_2)$, $x_1, x_2 \in_R \mathbb{Z}_n$,
  - $pk_P = (g^{x_1} = h_1, g^{x_2} = h_2)$.
- 1. $P : u_1 \in_R \mathbb{Z}_n, u_2 \in_R \mathbb{Z}_n$, $a_1 = g^{u_1}$, $a_2 = g^{u_2}$
  2. $P \to V : a_1, a_2$
  3. $V : c, \in_R \mathbb{Z}_n$
  4. $V \to P : c$
  5. $P : r_1 = u_1 + x_1 \cdot c, r_2 = u_2 + x_2 \cdot c$
  6. $P \to V : r_1, r_2$
  7. $V : g^{r_1} \stackrel{?}{=} a \cdot h_1^c, g^{r_2} \stackrel{?}{=} a \cdot h_2^c$

AND-композиция Σ-протоколов

- Пусть $\langle g \rangle$ — циклическая группа с порождающим элементом g, $n = |\langle g \rangle|$ – порядок этой группы. **Ключи доказывающей стороны**:
  - $sk_P = (x_1, x_2)$, $x_1, x_2 \in_R \mathbb{Z}_n$,
  - $pk_P = (g^{x_1} = h_1, g^{x_2} = h_2)$.
- 1. $P : u_1 \in_R \mathbb{Z}_n, u_2 \in_R \mathbb{Z}_n,\ a_1 = g^{u_1},\ a_2 = g^{u_2}$
  2. $P \to V : a_1, a_2$
  3. $V : c, \in_R \mathbb{Z}_n$
  4. $V \to P : c$
  5. $P : r_1 = u_1 + x_1 \cdot c, r_2 = u_2 + x_2 \cdot c$
  6. $P \to V : r_1, r_2$
  7. $V : g^{r_1} \stackrel{?}{=} a \cdot h_1^c, g^{r_2} \stackrel{?}{=} a \cdot h_2^c$

Утверждение

Изображенная AND-композиция для Σ-протоколов является Σ-протоколом для отношения $R = \{(h_1, h_2, x_1, x_2) : g^{x_1} = h_1, g^{x_2} = h_2\}$.

Доказательство:

AND-композиция Σ-протоколов

- Пусть $\langle g \rangle$ — циклическая группа с порождающим элементом g, $n = |\langle g \rangle|$ – порядок этой группы. **Ключи доказывающей стороны**:
  - $sk_P = (x_1, x_2)$, $x_1, x_2 \in_R \mathbb{Z}_n$,
  - $pk_P = (g^{x_1} = h_1, g^{x_2} = h_2)$.
- 1. $P : u_1 \in_R \mathbb{Z}_n, u_2 \in_R \mathbb{Z}_n,\ a_1 = g^{u_1},\ a_2 = g^{u_2}$
  2. $P \to V : a_1, a_2$
  3. $V : c, \in_R \mathbb{Z}_n$
  4. $V \to P : c$
  5. $P : r_1 = u_1 + x_1 \cdot c, r_2 = u_2 + x_2 \cdot c$
  6. $P \to V : r_1, r_2$
  7. $V : g^{r_1} \stackrel{?}{=} a \cdot h_1^c, g^{r_2} \stackrel{?}{=} a \cdot h_2^c$

Утверждение

Изображенная AND-композиция для Σ-протоколов является Σ-протоколом для отношения $R = \{(h_1, h_2, x_1, x_2) : g^{x_1} = h_1, g^{x_2} = h_2\}$.

Доказательство: Самостоятельно.

EQ-композиция Σ-протоколов

- Пусть $\langle g_1 \rangle = \langle g_2 \rangle$ — циклическая группа, g_1 и g_2 — разные порождающие элементы, $n = |\langle g_1 \rangle|$ – порядок этой группы.
  Ключи доказывающей стороны:
  - $sk_P = x$, $x \in_R \mathbb{Z}_n$,
  - $pk_P = (g_1^x = h_1, g_2^x = h_2)$.

EQ-композиция Σ-протоколов

- Пусть $\langle g_1 \rangle = \langle g_2 \rangle$ — циклическая группа, g_1 и g_2 — разные порождающие элементы, $n = |\langle g_1 \rangle|$ – порядок этой группы.
  Ключи доказывающей стороны:
  - $sk_P = x$, $x \in_R \mathbb{Z}_n$,
  - $pk_P = (g_1^x = h_1, g_2^x = h_2)$.
- 1. $P : u \in_R \mathbb{Z}_n$, $a_1 = g_1^u$, $a_2 = g_2^u$

EQ-композиция Σ-протоколов

- Пусть $\langle g_1 \rangle = \langle g_2 \rangle$ — циклическая группа, g_1 и g_2 — разные порождающие элементы, $n = |\langle g_1 \rangle|$ – порядок этой группы.
  Ключи доказывающей стороны:
  - $sk_P = x$, $x \in_R \mathbb{Z}_n$,
  - $pk_P = (g_1^x = h_1, g_2^x = h_2)$.
- 1. $P : u \in_R \mathbb{Z}_n$, $a_1 = g_1^u$, $a_2 = g_2^u$
  2. $P \to V : a_1, a_2$

EQ-композиция Σ-протоколов

- Пусть $\langle g_1 \rangle = \langle g_2 \rangle$ — циклическая группа, g_1 и g_2 — разные порождающие элементы, $n = |\langle g_1 \rangle|$ – порядок этой группы.
  Ключи доказывающей стороны:
  - $sk_P = x$, $x \in_R \mathbb{Z}_n$,
  - $pk_P = (g_1^x = h_1, g_2^x = h_2)$.
- 1. $P : u \in_R \mathbb{Z}_n$, $a_1 = g_1^u$, $a_2 = g_2^u$
  2. $P \to V : a_1, a_2$
  3. $V : c, \in_R \mathbb{Z}_n$

EQ-композиция Σ-протоколов

- Пусть $\langle g_1 \rangle = \langle g_2 \rangle$ — циклическая группа, g_1 и g_2 — разные порождающие элементы, $n = |\langle g_1 \rangle|$ – порядок этой группы.
  Ключи доказывающей стороны:
  - $sk_P = x$, $x \in_R \mathbb{Z}_n$,
  - $pk_P = (g_1^x = h_1, g_2^x = h_2)$.
- 1. $P : u \in_R \mathbb{Z}_n$, $a_1 = g_1^u$, $a_2 = g_2^u$
  2. $P \to V : a_1, a_2$
  3. $V : c, \in_R \mathbb{Z}_n$
  4. $V \to P : c$

EQ-композиция Σ-протоколов

- Пусть $\langle g_1 \rangle = \langle g_2 \rangle$ — циклическая группа, g_1 и g_2 — разные порождающие элементы, $n = |\langle g_1 \rangle|$ – порядок этой группы.
  Ключи доказывающей стороны:
  - $sk_P = x$, $x \in_R \mathbb{Z}_n$,
  - $pk_P = (g_1^x = h_1, g_2^x = h_2)$.
- 1. $P : u \in_R \mathbb{Z}_n$, $a_1 = g_1^u$, $a_2 = g_2^u$
  2. $P \to V : a_1, a_2$
  3. $V : c, \in_R \mathbb{Z}_n$
  4. $V \to P : c$
  5. $P : r = u + x \cdot c$

EQ-композиция Σ-протоколов

- Пусть $\langle g_1 \rangle = \langle g_2 \rangle$ — циклическая группа, g_1 и g_2 — разные порождающие элементы, $n = |\langle g_1 \rangle|$ – порядок этой группы.
  Ключи доказывающей стороны:
  - $sk_P = x$, $x \in_R \mathbb{Z}_n$,
  - $pk_P = (g_1^x = h_1, g_2^x = h_2)$.
- 1. $P : u \in_R \mathbb{Z}_n$, $a_1 = g_1^u$, $a_2 = g_2^u$
  2. $P \to V : a_1, a_2$
  3. $V : c, \in_R \mathbb{Z}_n$
  4. $V \to P : c$
  5. $P : r = u + x \cdot c$
  6. $P \to V : r$

EQ-композиция Σ-протоколов

- Пусть $\langle g_1 \rangle = \langle g_2 \rangle$ — циклическая группа, g_1 и g_2 — разные порождающие элементы, $n = |\langle g_1 \rangle|$ – порядок этой группы.
  Ключи доказывающей стороны:
  - $sk_P = x$, $x \in_R \mathbb{Z}_n$,
  - $pk_P = (g_1^x = h_1, g_2^x = h_2)$.
- 1. $P : u \in_R \mathbb{Z}_n$, $a_1 = g_1^u$, $a_2 = g_2^u$
  2. $P \to V : a_1, a_2$
  3. $V : c, \in_R \mathbb{Z}_n$
  4. $V \to P : c$
  5. $P : r = u + x \cdot c$
  6. $P \to V : r$
  7. $V : g_1^r \stackrel{?}{=} a_1 \cdot h_1^c, g^r \stackrel{?}{=} a_2 \cdot h_2^c$

EQ-композиция Σ-протоколов

- Пусть $\langle g_1 \rangle = \langle g_2 \rangle$ — циклическая группа, g_1 и g_2 — разные порождающие элементы, $n = |\langle g_1 \rangle|$ – порядок этой группы.
  Ключи доказывающей стороны:
  - $sk_P = x$, $x \in_R \mathbb{Z}_n$,
  - $pk_P = (g_1^x = h_1, g_2^x = h_2)$.
- 1. $P : u \in_R \mathbb{Z}_n$, $a_1 = g_1^u$, $a_2 = g_2^u$
  2. $P \to V : a_1, a_2$
  3. $V : c, \in_R \mathbb{Z}_n$
  4. $V \to P : c$
  5. $P : r = u + x \cdot c$
  6. $P \to V : r$
  7. $V : g_1^r \stackrel{?}{=} a_1 \cdot h_1^c, g^r \stackrel{?}{=} a_2 \cdot h_2^c$

Утверждение

Изображенная EQ-композиция для Σ-протоколов является Σ-протоколом для отношения
$R = \{(g_1, h_1, g_2, h_2, x) : g_1^x = h_1, g_2^x = h_2\}$.

Доказательство:

EQ-композиция Σ-протоколов

- Пусть $\langle g_1 \rangle = \langle g_2 \rangle$ — циклическая группа, g_1 и g_2 — разные порождающие элементы, $n = |\langle g_1 \rangle|$ – порядок этой группы.
  Ключи доказывающей стороны:
  - $sk_P = x,\ x \in_R \mathbb{Z}_n,$
  - $pk_P = (g_1^x = h_1, g_2^x = h_2).$
- 1. $P : u \in_R \mathbb{Z}_n,\ a_1 = g_1^u,\ a_2 = g_2^u$
  2. $P \to V : a_1, a_2$
  3. $V : c, \in_R \mathbb{Z}_n$
  4. $V \to P : c$
  5. $P : r = u + x \cdot c$
  6. $P \to V : r$
  7. $V : g_1^r \stackrel{?}{=} a_1 \cdot h_1^c, g^r \stackrel{?}{=} a_2 \cdot h_2^c$

Утверждение

Изображенная EQ-композиция для Σ-протоколов является Σ-протоколом для отношения
$R = \{(g_1, h_1, g_2, h_2, x) : g_1^x = h_1, g_2^x = h_2\}.$

Доказательство: Б/д.

OR-композиция Σ-протоколов

Пусть $\langle g \rangle$ — циклическая группа, g – порождающий элемент, $n = |\langle g \rangle|$ – порядок этой группы.

Ключи доказывающей стороны:

- или $sk_P = x_1$, $x_1 \in_R \mathbb{Z}_n$, или $sk_P = x_2$, $x_2 \in_R \mathbb{Z}_n$
- или $pk_P = g^{x_1} = h_1$, или $pk_P = g^{x_2} = h_2$ (оба ключа известны проверяющей стороне).

OR-композиция Σ-протоколов

Пусть $\langle g \rangle$ — циклическая группа, g – порождающий элемент, $n = |\langle g \rangle|$ – порядок этой группы.

Ключи доказывающей стороны:

- или $sk_P = x_1$, $x_1 \in_R \mathbb{Z}_n$, или $sk_P = x_2$, $x_2 \in_R \mathbb{Z}_n$
- или $pk_P = g^{x_1} = h_1$, или $pk_P = g^{x_2} = h_2$ (оба ключа известны проверяющей стороне).

P знает хотя бы x_1

1. $P : c_2, r_2, u_1 \in_R \mathbb{Z}_n$, $a_1 = g^{u_1}$, $a_2 = g^{r_2} h^{-c_2}$
2. $P \to V : a_1, a_2$
3. $V : c \in_R \mathbb{Z}_n$
4. $V \to P : c$
5. $P : c_1 = c - c_2, r_1 = u_1 + x_1 \cdot c_1$
6. $P \to V : c_1, c_2, r_1, r_2$
7. $V : g^{r_1} \stackrel{?}{=} a_1 \cdot h_1^{c_1}, g^{r_2} \stackrel{?}{=} a_2 \cdot h_2^{c_2}$

OR-композиция Σ-протоколов

Пусть $\langle g \rangle$ — циклическая группа, g – порождающий элемент, $n = |\langle g \rangle|$ – порядок этой группы.

Ключи доказывающей стороны:

- или $sk_P = x_1$, $x_1 \in_R \mathbb{Z}_n$, или $sk_P = x_2$, $x_2 \in_R \mathbb{Z}_n$
- или $pk_P = g^{x_1} = h_1$, или $pk_P = g^{x_2} = h_2$ (оба ключа известны проверяющей стороне).

P знает хотя бы x_1

1. $P : c_2, r_2, u_1 \in_R \mathbb{Z}_n$, $a_1 = g^{u_1}$, $a_2 = g^{r_2} h^{-c_2}$
2. $P \to V : a_1, a_2$
3. $V : c \in_R \mathbb{Z}_n$
4. $V \to P : c$
5. $P : c_1 = c - c_2, r_1 = u_1 + x_1 \cdot c_1$
6. $P \to V : c_1, c_2, r_1, r_2$
7. $V : g^{r_1} \stackrel{?}{=} a_1 \cdot h_1^{c_1}, g^{r_2} \stackrel{?}{=} a_2 \cdot h_2^{c_2}$

P знает хотя бы x_2

1. $P : c_1, r_1, u_2 \in_R \mathbb{Z}_n$, $a_1 = g^{r_1} h^{-c_1}$, $a_2 = g^{u_2}$
2. $P \to V : a_1, a_2$
3. $V : c \in_R \mathbb{Z}_n$
4. $V \to P : c$
5. $P : c_2 = c - c_1, r_2 = u_2 + x_2 \cdot c_2$
6. $P \to V : c_1, c_2, r_1, r_2$
7. $V : g^{r_1} \stackrel{?}{=} a_1 \cdot h_1^{c_1}, g^{r_2} \stackrel{?}{=} a_2 \cdot h_2^{c_2}$

OR-композиция Σ-протоколов. Утверждение

Утверждение

Изображенная на предыдущем слайде OR-композиция для Σ-протоколов является Σ-протоколом для отношения $R = \{(h_1, h_2, x_1, x_2) : g^{x_1} = h_1 \vee g^{x_2} = h_2\}$.

Доказательство:

OR-композиция Σ-протоколов. Утверждение

Утверждение

Изображенная на предыдущем слайде OR-композиция для Σ-протоколов является Σ-протоколом для отношения $R = \{(h_1, h_2, x_1, x_2) : g^{x_1} = h_1 \vee g^{x_2} = h_2\}$.

Доказательство: Б/д.

Отличие интерактивных протоколов от неинтерактивных

Схема аутентификации

Отличие интерактивных протоколов от неинтерактивных

Схема аутентификации

- **Интерактивная** – это схемы идентификации (проверяющий задает вопросы, доказывающий отвечает).

Отличие интерактивных протоколов от неинтерактивных

Схема аутентификации

- **Интерактивная** – это схемы идентификации (проверяющий задает вопросы, доказывающий отвечает).
- **Неинтерактивная** – это схемы цифровой подписи (проверяющий проверяет подпись, можно обойтись без вопросов, например, на основе метки времени).

Отличие интерактивных протоколов от неинтерактивных

Схема аутентификации

- **Интерактивная** – это схемы идентификации (проверяющий задает вопросы, доказывающий отвечает).
- **Неинтерактивная** – это схемы цифровой подписи (проверяющий проверяет подпись, можно обойтись без вопросов, например, на основе метки времени).

Схемы доказательства знания

Отличие интерактивных протоколов от неинтерактивных

Схема аутентификации

- **Интерактивная** – это схемы идентификации (проверяющий задает вопросы, доказывающий отвечает).
- **Неинтерактивная** – это схемы цифровой подписи (проверяющий проверяет подпись, можно обойтись без вопросов, например, на основе метки времени).

Схемы доказательства знания

- **Интерактивная**: схема включает протокол, в ходе которого доказывающий должен убедить проверяющую сторону в том, что он действительно знает, что некоторое заявление имеет место (утверждение выполняется).

Отличие интерактивных протоколов от неинтерактивных

Схема аутентификации

- **Интерактивная** – это схемы идентификации (проверяющий задает вопросы, доказывающий отвечает).
- **Неинтерактивная** – это схемы цифровой подписи (проверяющий проверяет подпись, можно обойтись без вопросов, например, на основе метки времени).

Схемы доказательства знания

- **Интерактивная**: схема включает протокол, в ходе которого доказывающий должен убедить проверяющую сторону в том, что он действительно знает, что некоторое заявление имеет место (утверждение выполняется).
- **Неинтерактивная**: схема включает **алгоритм генерации доказательства** утверждения и **алгоритм проверки доказательства**.

Отличие интерактивных протоколов от неинтерактивных. 2

Важное отличие

В неинтерактивных Σ-протоколах (или Σ-доказательствах, Σ-proofs) проверить правильность доказательства может любой участник (и в любое время).

Отличие интерактивных протоколов от неинтерактивных. 2

Важное отличие

В неинтерактивных Σ-протоколах (или Σ-доказательствах, Σ-proofs) проверить правильность доказательства может любой участник (и в любое время).

Цифровая подпись на основе Σ-протоколов – это пример преобразования интерактивного Σ-протокола в неинтерактивную версию.

Составляющие схемы ЭЦП и идея

Составляющие схемы ЭЦП и идея

- Алгоритм генерации ключей.

Составляющие схемы ЭЦП и идея

- Алгоритм генерации ключей.
- Алгоритм генерации подписи.

Составляющие схемы ЭЦП и идея

- Алгоритм генерации ключей.
- Алгоритм генерации подписи.
- Алгоритм проверки подписи.

Составляющие схемы ЭЦП и идея

- Алгоритм генерации ключей.
- Алгоритм генерации подписи.
- Алгоритм проверки подписи.
- Также предполагается, что имеется криптогр. хэш-функция $H : \{0,1\}^* \to \{0,1\}^k$.

Составляющие схемы ЭЦП и идея

- Алгоритм генерации ключей.
- Алгоритм генерации подписи.
- Алгоритм проверки подписи.
- Также предполагается, что имеется криптогр. хэш-функция $H : \{0,1\}^* \to \{0,1\}^k$.

Протокол Шнорра

1. $P : u \in_R \mathbb{Z}_n,\ a = g^u$
2. $P \to V : a$
3. $V : c \in_R \mathbb{Z}_n$
4. $V \to P : c$
5. $P : r = u + x \cdot c$
6. $P \to V : r$
7. $V : g^r \stackrel{?}{=} a \cdot h^c$

Составляющие схемы ЭЦП и идея

- Алгоритм генерации ключей.
- Алгоритм генерации подписи.
- Алгоритм проверки подписи.
- Также предполагается, что имеется криптогр. хэш-функция $H : \{0,1\}^* \to \{0,1\}^k$.

Протокол Шнорра

1. $P : u \in_R \mathbb{Z}_n,\ a = g^u$
2. $P \to V : a$
3. $V : c \in_R \mathbb{Z}_n$
4. $V \to P : c$
5. $P : r = u + x \cdot c$
6. $P \to V : r$
7. $V : g^r \stackrel{?}{=} a \cdot h^c$

Идея схемы подписи документа M, $sk_P = x$, $pk_P = g^x = h$.

1. $P : u \in_R \mathbb{Z}_n,\ a = g^u$
2. $P : c = H(a, M)$, $\{0, 1, ..., 2^k - 1\} \subset \mathbb{Z}_n$
3. $P : r = u + x \cdot c$
4. $P : \mathrm{SIGN}(M) = (c, r)$
5. $V : H(g^r h^{-c}, M) \stackrel{?}{=} c$

Составляющие схемы ЭЦП и идея

- Алгоритм генерации ключей.
- Алгоритм генерации подписи.
- Алгоритм проверки подписи.
- Также предполагается, что имеется криптогр. хэш-функция $H : \{0,1\}^* \to \{0,1\}^k$.

Протокол Шнорра

1. $P : u \in_R \mathbb{Z}_n,\ a = g^u$
2. $P \to V : a$
3. $V : c \in_R \mathbb{Z}_n$
4. $V \to P : c$
5. $P : r = u + x \cdot c$
6. $P \to V : r$
7. $V : g^r \stackrel{?}{=} a \cdot h^c$

Идея схемы подписи документа M, $sk_P = x$, $pk_P = g^x = h$.

1. $P : u \in_R \mathbb{Z}_n,\ a = g^u$
2. $P : c = H(a, M)$, $\{0, 1, ..., 2^k - 1\} \subset \mathbb{Z}_n$
3. $P : r = u + x \cdot c$
4. $P : \mathrm{SIGN}(M) = (c, r)$
5. $V : H(g^r h^{-c}, M) \stackrel{?}{=} c$

Эвристика Фиата-Шамира

Случайный выбор запроса c можно заменить вычислением криптографической хэш-функции от анонса и подписываемого текста.

Заключение

Спасибо за внимание!