Гибридизация ключей – однократного хэширования достаточно

Виталий Кирюхин

АНО «НТЦ ЦК», ООО «СФБ Лаб», АО «ИнфоТеКС»

РусКрипто’2025
20 марта 2025

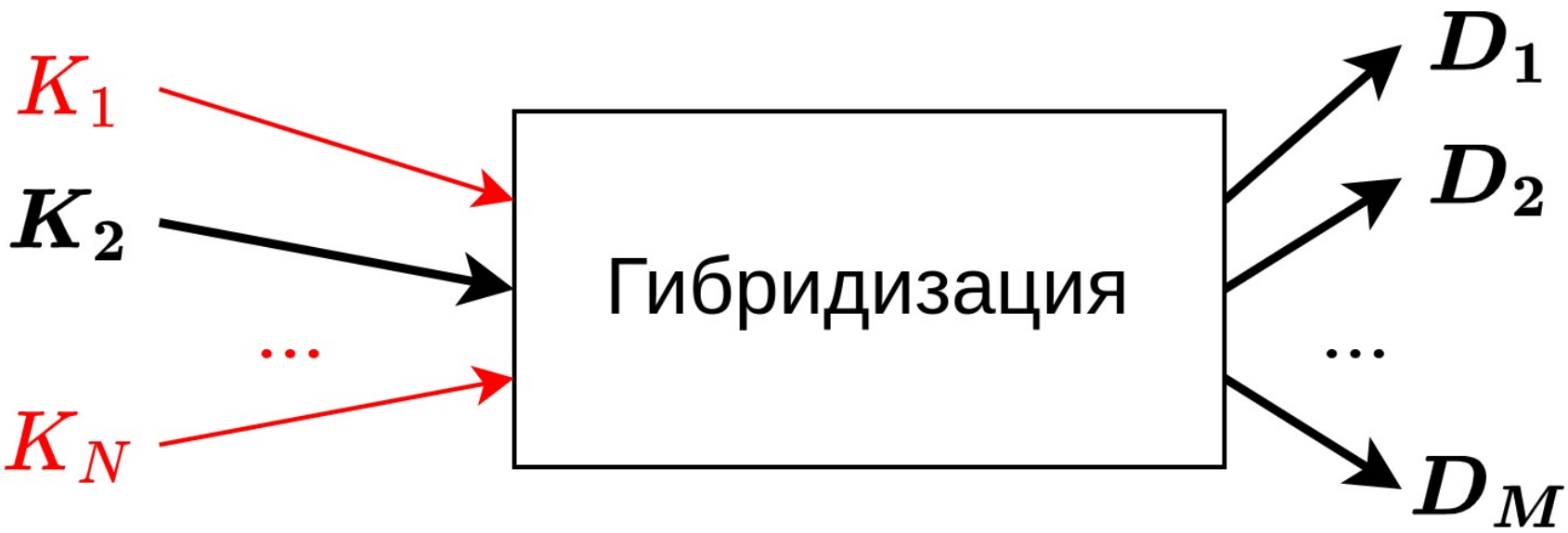
vitaly.kiryukhin@sfblaboratory.ru

Гибридизация

Алгоритмы гибридизации (комбайнеры) ключей формируют секретный производный ключ(и) из нескольких исходных (гибридизируемых).

Необходимость гибридизации

Параллельно используется *N* способов распределения ключей – «плохие» и **«хорошие»**. Система должна оставаться стойкой, если есть хотя бы один «**хороший**».



Возможные источники ключей

- Предварительное распределение
- Квантовое распределение (КРК)
- Классические/постквантовые асимметричные механизмы
- Различные комбинации перечисленного выше

Условия и требования

Цель: (вычислительная) *неотличимость* производных ключей от «идеальных», когда:

Условия и требования

Цель: (вычислительная) *неотличимость*

производных ключей от «идеальных», когда:

- хотя бы ***один*** из N ключей на входе ***секретный***,

  а остальные навязываются противником

Условия и требования

Цель: (вычислительная) *неотличимость*

производных ключей от «идеальных», когда:

- хотя бы ***один*** из *N* ключей на входе ***секретный***,

  а остальные навязываются противником

- неизвестно, какой именно из *N* ключей секретный

Условия и требования

Цель: (вычислительная) *неотличимость*
производных ключей от «идеальных», когда:

- хотя бы ***один*** из N ключей на входе ***секретный***,
  а остальные навязываются противником
- неизвестно, какой именно из N ключей секретный
- источник секретных ключей характеризуется
  малым вариационным расстоянием ε или
  малым преобладанием в некоторой задаче различения

Одноразовые и долговременные ключи

Простой случай

Гибридизируемые ключи **одноразовые**.

⇒ Секретный ключ тоже разовый –

НЕ гибридизируется повторно с *иными* наборами ключей.

Одноразовые и долговременные ключи

Простой случай

Гибридизируемые ключи **одноразовые**.

⇒ Секретный ключ тоже разовый –

НЕ гибридизируется повторно с *иными* наборами ключей.

Сложный случай

Среди гибридизируемых ключей есть **долговременный**.

⇒ Секретный ключ может быть долговременным и

может гибридизироваться с *иными* наборами ключей.

Одноразовые ключи

Одноразовые ключи

Для обеспечения стойкости достаточно XOR,

$$D = \mathrm{XOR}(K_1, \ldots, K_N) = K_1 \oplus \ldots \oplus K_N,$$

ключи K_1,...,K_N должны быть одинаковой длины.

Много производных

Если $M > 1$, то достаточно схемы XOR-then-PRF, к примеру,

$$D = \mathrm{XOR}(K_1, \ldots, K_N) = K_1 \oplus \ldots \oplus K_N,$$

$$D_1 = \mathrm{PRF}(D, 1),$$

$$D_2 = \mathrm{PRF}(D, 2),$$

$$\ldots$$

$$D_M = \mathrm{PRF}(D, M)$$

Недопустимость долговременных ключей у XOR

Пусть K_3 используется *дважды* в разных наборах ключей

$$D = \mathrm{XOR}(K_1 K_2, K_3) = K_1 \oplus K_2 \oplus K_3,$$
$$\tilde{D} = \mathrm{XOR}(\tilde{K}_1 \tilde{K}_2, K_3) = \tilde{K}_1 \oplus \tilde{K}_2 \oplus K_3.$$

Получившиеся ключи **связанные**.

Разность

$$\Delta = D \oplus \tilde{D}$$

известна атакующему или выбрана им.

Недопустимость долговременных ключей у XOR

Вывод

Если хотя бы один из N ключей
может использоваться повторно, то

- использование XOR *недопустимо*
- использование XOR-then-PRF *нежелательно*

Долговременный ключ

Долговременный ключ

Условия

Среди гибридизируемых ключей есть **долговременный**.

⇒ Секретный ключ может быть долговременным и

может гибридизироваться с *иными* наборами ключей.

Долговременный ключ

Условия

Среди гибридизируемых ключей есть **долговременный**.

$\Rightarrow$ Секретный ключ может быть долговременным и

может гибридизироваться с *иными* наборами ключей.

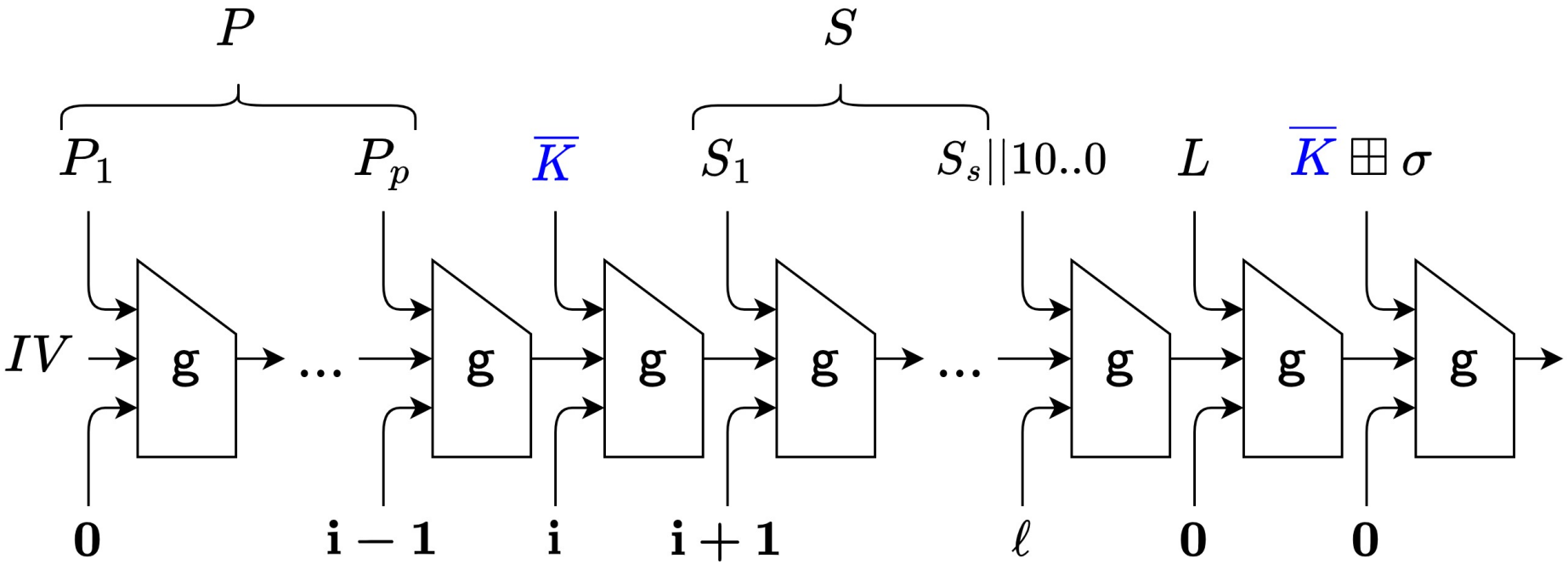
Пример

Один предраспределённый ключ $\boldsymbol{K}$.

L ключей $K^{(1)}$, ..., $K^{(L)}$ выработаны КРК.

L производных $D^{(1)} = \mathrm{Hybrid}(\boldsymbol{K}, K^{(1)})$, ..., $D^{(L)} = \mathrm{Hybrid}(\boldsymbol{K}, K^{(L)})$.

Решение – обобщённый «Ключевой Стрибог»



- Ключ $\overline{K} = K||0...0$ может находиться в *середине*
- Префикс P – произвольное число блоков
- Суффикс S – произвольное число бит
- Размер блока/состояния – $n = 512$ бит
- Никаких изменений в самой хэш-функции «Стрибог»

Решение – обобщённый «Ключевой Стрибог»

$$\mathrm{GKH}_\tau : V^k \times ((V^n)^{\leq p} \times V^*) \to V^\tau$$

$$\mathrm{GKH}_\tau(K, (P, S)) = \text{Стрибог}_\tau(P||\overline{K}||S)$$

- K – ε-секретный ключ длины k бит, $\overline{K} = K||0\ldots0$
- P – строка из произвольного числа блоков
- S – строка из произвольного числа бит
- $\tau \in \{256, 512\}$ – битовая длина выхода

Базовые задачи

Функция сжатия g должна быть:

1. Стойкой *PRF* (по обоим входам)

   при атаках со связанными ключами (*PRF-RKA*)
2. Стойкой к поиску коллизий (*TXCR*)
3. Стойкой к поиску прообраза (*TPR*)

Базовые задачи

Функция сжатия g должна быть:

1. Стойкой *PRF* (по обоим входам)

   при атаках со связанными ключами (*PRF-RKA*)
2. Стойкой к поиску коллизий (*TXCR*)
3. Стойкой к поиску прообраза (*TPR*)

Замечание

Свойство 1 необходимо для стойкости HMAC-Стрибог и Стрибог-K.

Свойства 2 и 3 не требуются в большинстве практических случаев.

PRF-стойкость

Теорема

$$\mathrm{Adv}^{PRF}_{\mathrm{GKH}}(t, q, \ell) \leq$$
$$\leq \varepsilon + \mathrm{Adv}^{TXCR}_{\mathrm{g}}(t', c^2) + \mathrm{Adv}^{TPR}_{\mathrm{g}}(t', c) + \mathrm{Adv}^{PRF\text{-}RKA_{\boxplus}}_{\mathrm{g}^{\triangledown}}(t', q', q', 1) +$$
$$+ q \cdot \ell \cdot \mathrm{Adv}^{PRF\text{-}RKA_{\oplus}}_{\mathrm{g}^{\triangleright}}(t', q, 2, 2) + \frac{3q^2}{2^n}$$

ε – вариационное расстояние, характеризующее ключ K

$t \approx t'$ – вычислительные ресурсы противника

q – число формируемых выходов, $q' = q + 1$

ℓ – общая длина хэшируемых данных, $c = \log_2(\ell)$

$n = 512$ – битовая длина состояния

PRF-стойкость

Эвристическая оценка: общий случай

$$\mathrm{Adv}^{PRF}_{\mathrm{GKH}} \lessapprox \varepsilon + \frac{t^2 \cdot \log_2^2(\ell)}{2^{n+1}} + \frac{t \cdot \log_2(\ell)}{2^n} + \frac{t}{2^k} + \frac{t \cdot q \cdot \ell}{2^{n-1}} + \frac{3q^2}{2^n}$$

ε – вариационное расстояние, характеризующее ключ K

t – вычислительные ресурсы противника

q – число формируемых выходов

ℓ – общая длина хэшируемых данных

$k \leq n$ – битовая длина ключа

$n = 512$ – битовая длина состояния

PRF-стойкость

Эвристическая оценка: общий случай

$$\mathrm{Adv}^{PRF}_{\mathrm{GKH}} \lessapprox \underbrace{\varepsilon}_{(1)} + \underbrace{\frac{t^2 \cdot \log_2^2(\ell)}{2^{n+1}}}_{(2)} + \underbrace{\frac{t \cdot \log_2(\ell)}{2^n}}_{(3)} + \underbrace{\frac{t}{2^k}}_{(4)} + \underbrace{\frac{t \cdot q \cdot \ell}{2^{n-1}}}_{(5)} + \underbrace{\frac{3q^2}{2^n}}_{(6)}$$

(1) – «Неидеальность» самого ключа K

(2) – Поиск коллизии

(3) – Поиск прообраза

(4) – Перебор секретного ключа

(5) – Деградация каскадного преобразования

(6) – Различные коллизии перед финализацией

PRF-стойкость: практические случаи

Эвристическая оценка: два ключа по 512 бит

$$\mathrm{Adv}^{PRF}_{\mathrm{GKH}} \lessapprox \varepsilon + \frac{t}{2^k} + \frac{t \cdot q \cdot \ell}{2^{n-1}} + \frac{3q^2}{2^n}$$

Эвристическая оценка: два ключа по 256 бит

$$\mathrm{Adv}^{PRF}_{\mathrm{GKH}} \lessapprox \varepsilon + \frac{t}{2^k} + \frac{t \cdot q \cdot \ell}{2^{n-1}} + \frac{3q^2}{2^n} \approx \varepsilon + \frac{t}{2^k}$$

PRF-стойкость: практические случаи

Эвристическая оценка: два ключа по 512 бит

$$\mathrm{Adv}^{PRF}_{\mathrm{GKH}} \lessapprox \varepsilon + \frac{t}{2^k} + \frac{t \cdot q \cdot \ell}{2^{n-1}} + \frac{3q^2}{2^n}$$

Эвристическая оценка: два ключа по 256 бит

$$\mathrm{Adv}^{PRF}_{\mathrm{GKH}} \lessapprox \varepsilon + \frac{t}{2^k} + \frac{t \cdot q \cdot \ell}{2^{n-1}} + \frac{3q^2}{2^n} \approx \varepsilon + \frac{t}{2^k}$$

Замечание

Здесь от g не требуется стойкость к коллизиям/прообразу.

KR-стойкость

Вероятность успешной атаки на 256-битный ключ

$$\mathrm{Adv}^{KR}_{\mathrm{GKH}}(t, q, \ell) \leq \mathrm{Adv}^{PRF}_{\mathrm{GKH}}(t', q+1, \ell) + \frac{1}{2^{\tau}} \lessapprox \varepsilon + \frac{t}{2^{k}}$$

t – вычислительные ресурсы противника

q – число формируемых выходов

ℓ – общая длина хэшируемых данных

$k = 256$ – длина ключа

KR-стойкость

Вероятность успешной атаки на 256-битный ключ

$$\mathrm{Adv}^{KR}_{\mathrm{GKH}}(t, q, \ell) \leq \mathrm{Adv}^{PRF}_{\mathrm{GKH}}(t', q+1, \ell) + \frac{1}{2^{\tau}} \lessapprox \varepsilon + \frac{t}{2^{k}}$$

t – вычислительные ресурсы противника

q – число формируемых выходов

ℓ – общая длина хэшируемых данных

$k = 256$ – длина ключа

$\Rightarrow$ Отсутствуют атаки лучше перебора

KR-стойкость

Вероятность успешной атаки на 512-битный ключ

$$\mathrm{Adv}^{KR}_{\mathsf{F}}(t, q, \ell) \le \mathrm{Adv}^{KR\text{-}RKA_{\circledast}}_{\mathsf{g}^{\triangledown}}(t', \tilde{q}) \lessapprox \varepsilon + \frac{t \cdot \tilde{q}}{2^k}$$

Алгоритм F	$\tilde{q}$	*Операция* $\circledast$
HMAC-Стрибог	$2 + 2q$	$\boxplus \circ \oplus$
Стрибог-K	$q + 1$	$\boxplus$
Обобщ. Стрибог-K	$2q$	$\boxplus$

KR-стойкость

Вероятность успешной атаки на 512-битный ключ

$$\mathrm{Adv}^{KR}_{\mathsf{F}}(t, q, \ell) \le \mathrm{Adv}^{KR\text{-}RKA_{\circledast}}_{\mathsf{g}^{\triangledown}}(t', \tilde{q}) \lessapprox \varepsilon + \frac{t \cdot \tilde{q}}{2^k}$$

Алгоритм F	$\tilde{q}$	*Операция* $\circledast$
HMAC-Стрибог	$2 + 2q$	$\boxplus \circ \oplus$
Стрибог-K	$q + 1$	$\boxplus$
Обобщ. Стрибог-K	$2q$	$\boxplus$

$\Rightarrow$ Стойкость деградирует не более чем линейно по q

Атаки на 512-битный ключ

Алгоритм	*Время* (t)	*Данные* $(\sigma \leq q\ell)$	*Ссылка*
Стрибог-С	2^{512}	—	SibeCrypt'23
HMAC-Стрибог	2^{419}	2^{419}	CRYPTO 2014
Стрибог-К	2^{419}	2^{419}	CTCrypt'22
Обобщ. Стрибог-К	2^{388}	2^{388}	**RusCrypto'25**
Теор. предел	2^{256}	2^{256}	

Схема гибридизации

Используем обобщённый «Ключевой Стрибог»,

$$\text{HybHash}_\tau(K_1, K_2, ..., K_N, prm) = \text{Стрибог}_\tau(\overline{K}_1||\overline{K}_2||...||\overline{K}_N||prm||ctx)$$

- Ключи дополняются нулями, $\overline{K}_i = K_i||0...0$
- Длины ключей и их число не ограничиваются
- Строка prm является опциональной (метки, счётчики и т.д.)
- Строка ctx кодирует число ключей N и их длины $|K_1|$, ..., $|K_N|$

Дополнение ключей

Общий случай

Может изменяться число ключей и/или их длины.

⇒ Каждый ключ дополняется нулями до 512 бит.

Упрощённый вариант

Число ключей постоянно, длина каждого не более 256 бит.

⇒ Каждый ключ дополняется нулями до 256 бит.

Число и длина ключей могут варьироваться

Пример для общего случая

Предварительное распределение: ключ $\boldsymbol{K}$ длины 256 бит

КРК: два ключа Q_1 и Q_2 по 512 бит

АКЕ-протокол: ключ A длины 384 бита

Пусть $\boldsymbol{K}$ секретный, а остальные навязаны, тогда допустимо

$$
\begin{aligned}
D_1 &= \mathrm{HybHash}(Q_1, A, \boldsymbol{K}) &&= \text{Стрибог}(Q_1 || A || 0^{128} || \boldsymbol{K} || 0^{256}) \\
D_2 &= \mathrm{HybHash}(\boldsymbol{K}, Q_2) &&= \text{Стрибог}(\boldsymbol{K} || 0^{256} || Q_2) \\
D_3 &= \mathrm{HybHash}(Q_1, \boldsymbol{K}, prm_1) &&= \text{Стрибог}(Q_1 || \boldsymbol{K} || 0^{256} || prm_1) \\
D_4 &= \mathrm{HybHash}(Q_1, \boldsymbol{K}, prm_2) &&= \text{Стрибог}(Q_1 || \boldsymbol{K} || 0^{256} || prm_2)
\end{aligned}
$$

Стойкость HybHash

Оценки снизу (по доказательству) для общего случая

1. Стойкость к атакам различения $\min(256, k)$ бит

Стойкость HybHash

Оценки снизу (по доказательству) для общего случая

1. Стойкость к атакам различения $\min(256, k)$ бит
2. Стойкость к атакам на ключ:
   - 2.1 k бит, при $k \leq 256$
   - 2.2 $\max(256, k - \log_2(q))$ бит, при $256 \leq k \leq 512$
   - 2.3 $\max(256, 512 - \log_2(q))$ бит, при $512 < k$

Общие свойства HybHash

1. Практически неограниченная нагрузка на ключ

Общие свойства HybHash

1. Практически неограниченная нагрузка на ключ
2. Вычислительная эффективность – от 4-х вызовов функции сжатия (9 у HMAC-Стрибог-512, 18 у HKDF-Стрибог-512)

Общие свойства HybHash

1. Практически неограниченная нагрузка на ключ
2. Вычислительная эффективность – от 4-х вызовов функции сжатия (9 у HMAC-Стрибог-512, 18 у HKDF-Стрибог-512)
3. Доказательство в «стандартной модели» – адаптируется на случай наличия у противника квантового вычислителя (Q1)

Ключевой материал вместо ключей

Ключевой материал

- Источник ключей **K** характеризуется

  малым вариационным (статистическим) расстоянием

$$\varepsilon = \frac{1}{2} \sum_{K \in V^k} |\Pr(\mathbf{K} = K) - 2^{-k}|$$

  или малым преобладанием в некоторой задаче различения

Ключевой материал

- Источник ключей **K** характеризуется

  малым вариационным (статистическим) расстоянием

$$\varepsilon = \frac{1}{2} \sum_{K \in V^k} |\Pr(\mathbf{K} = K) - 2^{-k}|$$

  или малым преобладанием в некоторой задаче различения

- Источник *ключевого материала* **M** характеризуется

  только минимальной энтропией

$$m = -\log_2 \max_M \Pr(\mathbf{M} = M)$$

Один вход – один выход

Схемой HybHash может быть сформировано из одного:

- ε-секретного ключа – неограниченное число выходов
- высокоэнтропийного материала – **один** выход

Универсальная хэш-функция

AU – Almost Universal Hash Function

$$\mathrm{Adv}_{\mathrm{F}}^{AU} = \max_{M,M',M\neq M'} \Pr(S \xleftarrow{\mathrm{R}} \mathbb{S} : \mathrm{F}(S,M) = \mathrm{F}(S,M'))$$

Стрибог$_\tau$ – (почти) универсальная хэш-функция, если несекретным значением «соли» S умозрительно считать раундовые константы функции сжатия.

Теорема

$$\mathrm{Adv}_{\text{Стрибог}_\tau}^{AU} \leq (l+1)\cdot \mathrm{Adv}_{\mathrm{g}}^{AU} + 2^{-\tau} \lessapprox (l+1)\cdot 2^{-512} + 2^{-\tau} = 2^{-\delta} + 2^{-\tau}$$

Экстрактор энтропии – «Leftover Hash Lemma»

Пусть $M_1,...,M_q$ – набор из различного ключевого материала, $m_i \geq m$, тогда совокупность производных ключей $D_1, ..., D_q$ характеризуется вариационным расстоянием

$$\varepsilon \leq \frac{1}{2} q \sqrt{2^{\tau}(2^{-m} + 2^{-\delta})}$$

Ограничения:

- мин-энтропия m существенно больше длины выхода τ
- вероятность коллизии $(2^{-\delta} + 2^{-\tau})$ близка к $2^{-\tau}$, выход хэш-функции строго меньше $n = 512$ бит

Все остальные случаи

- Низкоэнтропийные данные
- Один высокоэнтропийный вход формирует несколько производных ключей
- Иные способы форматирования входа

Все остальные случаи

- Низкоэнтропийные данные
- Один высокоэнтропийный вход формирует несколько производных ключей
- Иные способы форматирования входа

$\Rightarrow$ Используем гипотезу «Стрибог $\approx$ случайный оракул»

- адекватное приближение при $t \ll 2^{\frac{n}{2}}$
- базовое предположение – «идеальный шифр»
- не адаптируется для случая квантовых вычислений

Akhmetzyanova L., Babueva A., Bozhko A., Smyshlyaev S.

Streebog as a Random Oracle

CTCrypt 2023

Заключение

Заключение

1. Алгоритмы гибридизации повышают стойкость при использовании нескольких систем распределения ключей

Заключение

1. Алгоритмы гибридизации повышают стойкость при использовании нескольких систем распределения ключей
2. XOR допустимо использовать только если все гибридизируемые ключи одноразовые

Заключение

1. Алгоритмы гибридизации повышают стойкость при использовании нескольких систем распределения ключей
2. XOR допустимо использовать только если все гибридизируемые ключи одноразовые
3. Схема **HybHash** (однократное хэширование «Стрибогом») – высокая стойкость в разнообразных сценариях

Заключение

1. Алгоритмы гибридизации повышают стойкость при использовании нескольких систем распределения ключей
2. XOR допустимо использовать только если все гибридизируемые ключи одноразовые
3. Схема **HybHash** (однократное хэширование «Стрибогом») – высокая стойкость в разнообразных сценариях
   3.1 Один ε-секретный ключ можно использовать неоднократно

Заключение

1. Алгоритмы гибридизации повышают стойкость при использовании нескольких систем распределения ключей
2. XOR допустимо использовать только если все гибридизируемые ключи одноразовые
3. Схема **HybHash** (однократное хэширование «Стрибогом») – высокая стойкость в разнообразных сценариях
   - 3.1 Один ε-секретный ключ можно использовать неоднократно
   - 3.2 Длины гибридизируемых ключей и их число могут меняться

Заключение

1. Алгоритмы гибридизации повышают стойкость при использовании нескольких систем распределения ключей
2. XOR допустимо использовать только если все гибридизируемые ключи одноразовые
3. Схема **HybHash** (однократное хэширование «Стрибогом») – высокая стойкость в разнообразных сценариях
   - 3.1 Один ε-секретный ключ можно использовать неоднократно
   - 3.2 Длины гибридизируемых ключей и их число могут меняться
   - 3.3 Практически неограниченная нагрузка на ключ

Заключение

1. Алгоритмы гибридизации повышают стойкость при использовании нескольких систем распределения ключей
2. XOR допустимо использовать только если все гибридизируемые ключи одноразовые
3. Схема **HybHash** (однократное хэширование «Стрибогом») – высокая стойкость в разнообразных сценариях
   - 3.1 Один ε-секретный ключ можно использовать неоднократно
   - 3.2 Длины гибридизируемых ключей и их число могут меняться
   - 3.3 Практически неограниченная нагрузка на ключ
   - 3.4 «Идеальная стойкость» к атакам на ключ/различение при 256-битных ключах

Заключение

1. Алгоритмы гибридизации повышают стойкость при использовании нескольких систем распределения ключей
2. XOR допустимо использовать только если все гибридизируемые ключи одноразовые
3. Схема **HybHash** (однократное хэширование «Стрибогом») – высокая стойкость в разнообразных сценариях
   - 3.1 Один ε-секретный ключ можно использовать неоднократно
   - 3.2 Длины гибридизируемых ключей и их число могут меняться
   - 3.3 Практически неограниченная нагрузка на ключ
   - 3.4 «Идеальная стойкость» к атакам на ключ/различение при 256-битных ключах
   - 3.5 Служит экстрактором энтропии при использовании ключевого материала вместо ключей

Благодарю за внимание!

Виталий Кирюхин
АНО «НТЦ ЦК», ООО «СФБ Лаб», АО «ИнфоТеКС»
РусКрипто’2025
20 марта 2025
vitaly.kiryukhin@sfblaboratory.ru