# Хэш-функции

Хэш-функция – преобразовывает (свертывает) массив произвольной (или почти произвольной) длины в строку установленной длины.

Криптографическая хэш-функция – хэш-функция, которая может использоваться для надежного шифрования данных. Она должна удовлетворять требованиям:

* Необратима (отсутствует полиномиальный алгоритм вычисления обратной функции) – невозможно по свертке установить исходный массив за реальное время.
* Не позволяет, зная массив и его свертку, найти другой массив с такой же сверткой за реальное время.
* Не позволяет эффективно искать коллизии (разные массивы с одинаковыми хэшами).
* (дополнительно) Лавинный эффект – небольшое изменение аргумента вызывает сильное изменение хэша.

Существует 2 группы хэш-функций:

* Быстрые на железе (SHA2/3, MD4/5) – быстрые, но не слишком стойкие – не подходят для хранения паролей, но подходят для хэширования «в реальном времени».
* Медленные на железе (Bcrypt, scrypt, Argon2) – требуют много времени или памяти, зато очень эффективные – оптимальны для хранения особо ценной информации.

SHA-3 (и более ранние SHA-1 и SHA-2 и их модификации) – SHA3 вроде ок

MD5 (MD4) – оба не рекомендуются из-за слабости

[ГОСТ Р 34.11-2012 (Стрибог)](https://ru.wikipedia.org/wiki/%D0%93%D0%9E%D0%A1%D0%A2_%D0%A0_34.11-2012) – Меркл-Дамгор + MD. Кузнечик?

Bcrypt (<https://ru.wikipedia.org/wiki/Bcrypt>)

Scrypt (<https://www.tarsnap.com/scrypt.html>)

Argon2 (выбрано в рамках Password Hashing Competition <https://password-hashing.net/>)

Tiger

RIPEMD-160

# Построение хэш-функций

Через доказательство (доказать, что задача поиска коллизий может быть сведена за полиномиальное время к доказуемо неполиномиальной задаче).

Примеры неразрешимых за полиномиальное время задач:

* [Дискретное логарифмирование](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D1%81%D0%BA%D1%80%D0%B5%D1%82%D0%BD%D0%BE%D0%B5_%D0%BB%D0%BE%D0%B3%D0%B0%D1%80%D0%B8%D1%84%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)
* [Нахождение квадратичного вычета](https://ru.wikipedia.org/wiki/%D0%9A%D0%B2%D0%B0%D0%B4%D1%80%D0%B0%D1%82%D0%B8%D1%87%D0%BD%D1%8B%D0%B9_%D0%B2%D1%8B%D1%87%D0%B5%D1%82)
* [Факторизация целых чисел](https://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%BA%D1%82%D0%BE%D1%80%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F_%D1%86%D0%B5%D0%BB%D1%8B%D1%85_%D1%87%D0%B8%D1%81%D0%B5%D0%BB)
* [Задача о сумме подмножеств](https://ru.wikipedia.org/wiki/%D0%97%D0%B0%D0%B4%D0%B0%D1%87%D0%B0_%D0%BE_%D1%81%D1%83%D0%BC%D0%BC%D0%B5_%D0%BF%D0%BE%D0%B4%D0%BC%D0%BD%D0%BE%D0%B6%D0%B5%D1%81%D1%82%D0%B2)

# Атаки на хэш-функции

Атака дней рождения – если длина хэша – n, то для поиска 1 коллизии с вероятностью более 50% будет достаточно 2^(n/2) переборов (вместо 2^n). Если выходы распределены неравномерно, то даже быстрее.

Другие атаки:

* построение мультиколлизий;
* удлинение прообраза;
* дифференциальный криптоанализ;
* линейный криптоанализ
* на основе фиксированных точек
* на основе связанных и разностных связанных ключей
* the Rebound attack = rotation cryptanalysis + дифференциальный криптоанализ
* атака дней рождения