**Хэширование** – преобр. вх. массива данных опр. типа произв. длины в вых. бит. строку фикс. длины: *XOR (длина всегда 8 бит)*

**Хэш-функция** – мат. ф, h=H(M), приним. на входе строку символов М (прообраз) перем. длины n, и преобраз. ее в вых. строку фикс. длины *l* (*хэш, дайджест, хэш-таблица*)  
1) бесключевые – завис. т. от сообщения 2) с секр. ключом

**Криптогр. хэш-функция** – спец. класс хеш-ф-й, имеет разл. св-ва для решения крипт. задач:  
***Задачи:*** для аутент (хран. паролей), проверка целостности д-х, защита файлов, обнаруж. зловред. ПО, криптовалют. технологии

**Свойства хэш-функций:**\* детерминированность (неважно ск. раз вычисл H(M), M – const, один. алг == один h)  
\* дб быстрым  
\* односторонность преобразования  
\* мин. изменения в M≠M’ изм. в H(M) ≠H(M’) -->иначе **Коллизия хэш-функции**\* коллизионная устойчивость: зная M, трудно найти M’, чтобы H(M)=H(M’) – иначе I рода  
 если случайно выбраны два сообщ (M, M’) для кот. мы узнаем хешH(M)=H(M’), II рода

**Типы хэш-ф-й:**  
\* специализированные (md2, md4, md5, sha…)  
\* на основе блоч. шифров (mdc-2, mdc-4)  
\* на основе модуляр. арифметики

Юзер хочет восст. пароль, это возможно т. если с-ма не исп. хэширование, или с-ма исп. БД, кот. содержит простые частые пароли (123456 или qwerty) – Online Reverse Hash Lookup

**Хэш-функции на основе блочных шифров:**

блоч. шифр необратим по ключу шифрования, и если исп. в ка-че ключа выход пред. шага, а в кач-ве Mi очередной блок сообщ, м. получить хорошую хэш-ф. (ГОСТ)  
НО: низкая производительность

**Мера криптостойкости** – вычислительная сложность нахождения коллизии, средство поиска коллизий – «парадокс дня рождения»

**Парадокс «дней рождений»:**

***Постулат:*** *в группе из 23+ чел с вероятность 0,5+др у двух одинаковое (нет близнецов, люди рожд. независимо, равномерно, случайно)*

P(An) – вер, что в группе из n чел нет один. др  
P(A2) = 364/365  
P(A3) = (364/365)\*(363/365)  
P(A23) = P(A22)\* (365-22)/365 = 0,493 => Pc(A23) = 0,507  
P(An) = P(An-1) \* (m-(n-1)) / m

**Хеш-функция** – ф., вып. отображение из мн-ва М в интервале [0,m-1]: h: M-->[0,m-1]

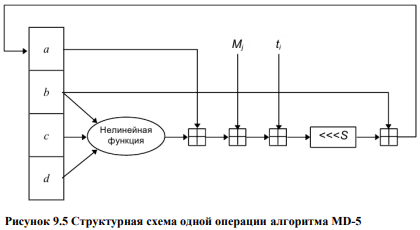
**Атака:**

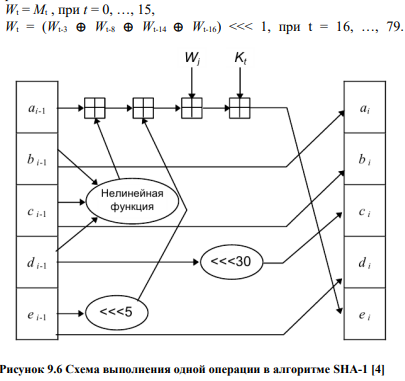
**m** – кол-во дней, **M** – мн-во людей группы, **n** – кол-во людей, люди хэшируются в их др (одно из m)  
стоит задача отыскание наим. n, при к-м хеши двух значений m будут «одним др»:  
n = (2m\* ln 100)1/2

**Особенности хэш-функций:**

**MD** – Message Digest – 6 не стандартиз, преобр. вх. сообщ. [max 263] в хеш [128], в MD6 хеш[1-512]  
**SHA** – Secure Hash Algorithm – max длина сообщ как в MD, но длина хешей разная  
**ГОСТ** – России   
**SHA3** – исп. «*криптогр. губку*» - итерат. подход с произв. длиной на входе и выходе на осн. преобразования (в РБ)

 **Алгоритмы MD и SHA:**  
\* расшир вх. сообщ длиной L (итог. длина L’ кратна 512, в незаполн. 64 бита запис. двоич. L)  
 1 бит доб. к сообщ, затем доб. биты 0 … в конце длина L  
\* разбивка расшир. сообщ на блоки   
\* иниц. нач. констант  
\* обработка сообщения поблочно  
\* вывод рез.





**SHA-256:** Исходное сообщение после дополнения разбивается на блоки, каждый блок — на 16 слов.   
Алгоритм пропускает каждый блок сообщения через цикл с 64 итерациями.   
На каждой итерации 2 слова преобразуются, функцию преобразования задают остальные слова.   
Результаты обработки каждого блока складываются, сумма является значением хеш-функции.   
Так как инициализация внутреннего состояния производится результатом обработки предыдущего блока, то нет возможности обрабатывать блоки параллельно.