УДК 004.424.47.021

СРАВНИТЕЛЬНЫЙ АНАЛИЗ МЕТОДОВ ХЕШИРОВАНИЯ

Кабыш Я.А.
Научный руководитель – Воронич Л.В., ассистент

В настоящее время в связи со стремительным развитием информационных технологий возрастает необходимость в обеспечении надежности и целостности передаваемых и хранимых данных. Одним из инструментов решения данных проблем стало хеширование информации, оказавшееся весьма востребованным в различных сферах, связанных как с малыми, так и с большими объемами данных.

Хеширование (англ. hashing) – это преобразование входного массива данных любой данных в выходную строку данных заданной длины. Осуществляется это при помощи хеш-функций (функций свертки) – специальных алгоритмов, которые в качестве входного параметра получают данные, состоящие из любого количества бит, а в качестве выходным данных выдают данные строго фиксированной длины, которые называются хешем. Процесс хеширования представлен на рис. 1.

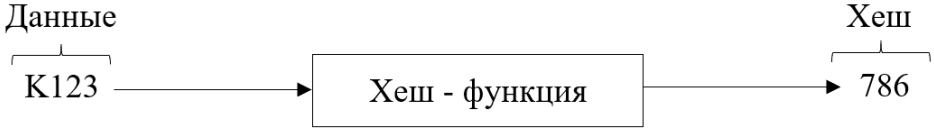

Рис.1. Процесс хеширования

Хеш-функции имеют некоторые базовые требования, такие как:

– Полученный хеш должен всегда быть фиксированного размера;
– Все одинаковые входные данные должны всегда приводиться к одинаковому хешу;
– Все разные входные данные должны приводить к разному хешу;
– Преобразование данных должно быть необратимо.

Однако в силу ограниченности возможной длины выходных данных может происходить ситуация, при которой двум разным входным значениям будет соответствовать одно и тоже значения хеша. Такая ситуация называется коллизией.

Существуют различные методы хеширования, такие как:

MD (Message Digest) – это семейство алгоритмов хеширования, таких как MD-2, MD-5 и другие, использовавшееся ранее во многих приложениях для генерации хеш-значений, однако в данный момент считается небезопасным из-за возможных коллизий. К примеру, алгоритмы MD-4 и MD-5 обладают следующими характеристиками:

– Сложность: являются относительно простым алгоритмом, использующий серию логических операций. Сложность в основном определяется длиной входных данных и характеристиками используемого оборудования;
– Скорость выполнения: обладают высокой скоростью выполнения на современном оборудовании и программном обеспечении;
– Цели применения: применялись для проверки целостности данных, аутентификации, генерации хеш-сумм и создания цифровых подписей, однако из-за уязвимостей и возможных коллизий не рекомендуется к использованию.

SHA (Secure Hash Algorithm) – это семейство криптографических алгоритмов хеширования, в число которых входят SHA-1, SHA-2, SHA-256 и другие. Более длинные версии обеспечивают более высокий уровень безопасности и стойкости к атакам. SHA является одним из наиболее широко используемых алгоритмов хеширования в сфере информационной безопасности. Эти алгоритмы обладают следующими характеристиками:
– Сложность: они основаны на различных логических и арифметических операциях, таких как битовые операции, циклические сдвиги, а также константные и нелинейные функции. В зависимости от версии сложность алгоритма может быть разной, но в общем случае она выше, чем у алгоритма MD5;
– Скорость выполнения: несмотря на более высокую сложность, обеспечивают высокую скорость выполнения на современном оборудовании;
– Цели применения: широко применяется в криптографических приложениях и протоколах для гарантирования целостности данных, создания цифровых подписей, аутентификации и ряде других задач. SHA также используется в качестве стандарта безопасности во многих криптографических протоколах, таких как TLS/SSL, IPsec, SSH.

CRC (Cyclic Redundancy Check) – это метод проверки целостности данных, используемый для обнаружения ошибок в данных после их передачи. Он основан на математическом аппарате полиномов и операциях деления с остатком. Принцип заключатся в вычислении контрольной суммы для набора данных, которая передается вместе с данными, после получения которых контрольная сумма находится повторно и сравнивается с переданной. Несовпадение сумм указывает на наличие ошибки, что требует повторной передачи данных. Основные характеристики алгоритма:
– Сложность: алгоритм прост в реализации и вычислительно не требователен, однако выбор подходящего полинома может потребовать некоторых ресурсов вычисления;

– Скорость выполнения: обеспечивает высокую скорость выполнения на современном оборудовании, так как требует лишь нескольких простых операций на каждый байт данных;
– Цели применения: основная цель алгоритма - обеспечение обнаружения ошибок в передаче данных или в хранимых файлах. Используется в таких протоколах, как Ethernet.

Таким образом, можно сделать следующие выводы:

– Алгоритмы MD5 и MD4, хотя и широко использовались в прошлом, сейчас считаются устаревшими и небезопасными из-за возможности коллизий и других уязвимостей. SHA, особенно более длинные версии, такие как SHA-256 и SHA-512, обеспечивают более высокий уровень безопасности и стойкости к атакам;
– Алгоритмы MD-5 и SHA в основном используются для обеспечения целостности данных, создания цифровых подписей и аутентификации, в то время как алгоритм CRC необходим для нахождения ошибок при передаче данных;
– Для задач, где требуется высокий уровень безопасности, рекомендуется использовать алгоритмы SHA, так как они обеспечивают более стойкую защиту данных. Алгоритм CRC подходит для обнаружения случайных ошибок в передаче данных. Алгоритм MD-5 не рекомендуется к использованию.

Исходя из всего вышеперечисленного, можно сделать вывод, что хеширование уже сейчас является неотъемлемой частью современных информационных технологий и может использоваться в различных базах данных для надежного хранения информации, передачи данных и другого, а необратимость хеш-функций делает их весьма сложными для взлома.

Литература

2. Бабаш, А. В. История криптографии. Часть I / А.В. Бабаш, Г.П. Шанкин. - М.: Гелиос АРВ, **2016**. - 240 с.
3. Жук А. П. Защита информации: Учебное пособие / А.П. Жук, Е.П. Жук, О.М. Лепешкин, А.И. Тимошкин. - 2-е изд. - М.: ИЦ РИОР: НИЦ ИНФРА-М, 2015. - 392 с.