УДК 681.324

В.Н. ГУГНИН, ст. преп. НТУ «ХПИ» (г. Харьков),
Д.В. СОТНИК, инж.-программист КП СПКБ АСУВ (г. Харьков)

КРИПТОАНАЛИЗ MD5

В статті розглядаються сучасні методи кріптоаналізу хеш-функції MD5 з метою розкриття паролю та підробки цифрового підпису документу. Зроблені висновки відповідно до перспектив кріптоаналізу на сучасному технічному рівні та наведені рекомендації до подальшого використання MD5 з урахуванням знайдених вразливостей.

In present article were overviewed methods of MD hash-function cryptanalysis for document digital signature falsification and password disclosing purposes. Conclusions regarding cryptanalysis aspects on modern technical level are given. Adducted recommendations for further MD using with taking in consideration of found weaknesses are given.

Постановка проблемы. Хэш-функции являются необходимым элементом многих криптографических схем. Под этим термином понимаются функции, отображающие сообщения произвольной длины (иногда длина сообщения ограничена, но достаточно большим числом) в значения фиксированной длины. Также их часто называют хэш-кодами. Таким образом, у всякой хэш-функции имеется большое количество коллизий, т.е. пар значений $x \neq y$ таких, что $h(x) = h(y)$.

Схемы электронной подписи – основная сфера применения хэш-функций в криптографии. Основное требование, предъявляемое к хэш-функциям в этом случае, состоит в отсутствии эффективных алгоритмов поиска коллизий. Поскольку используемые на практике схемы электронной подписи не приспособлены для подписания сообщений произвольной длины, а процедура, состоящая в разбиении сообщения на блоки и в генерации подписи для каждого блока по отдельности, крайне неэффективна, единственным разумным решением представляется применение схемы подписи к хэш-коду сообщения. Нетрудно понять, что наличие эффективных методов поиска коллизий для хэш-функции подрывает стойкость протокола электронной подписи [1].

Другой областью использования хэш-функции MD5 является необратимое шифрование паролей [2]. В этом случае основным требованием к хэш-функции становится невозможность определения исходного ключа по значению его хэш-кода. В настоящее время не существует алгоритмов, которые позволили бы восстановить исходный или эквивалентный ключ по его хэш-коду MD5 или выделить множество возможных ключей. Таким образом, поиск ключа сводится к прямому перебору и среднее количество необходимых попыток оценивается как 2^{80}, что делает этот метод практически не применимым. Однако, из-за возможности существования коллизий, перебор может

завершиться до нахождения исходного ключа подбором эквивалентного значения.

Таким образом, существование коллизий делает хеш-функцию уязвимой к прямому перебору при атаке на раскрытие ключа, а наличие эффективных методов поиска коллизий позволяет использовать их для подделки электронной подписи [3].

Анализ литературы. Хэш-функция MD5 была создана в 1992 г. на основании MD4 и ее безопасность была исследована большим количеством криптоаналитиков.

Первая публикация об уязвимости MD5, приводившей к созданию псевдоколлизий (для разных наборов инициализирующих переменных) датируется 1993 г. [4].

О существовании метода поиска коллизий впервые было заявлено 17 августа 2004 г. профессором Ван Сяоюнь (Wang Xiaoyun), университет Шандонг (Shandong University). Предложенный метод позволял определить ключ, эквивалентный исходному (коллизию), за 2^{39} и 2^{32} операций хеширования для первого и второго блока ключа соответственно, вероятность нахождения коллизии возросла с 2^{-37} до 2^{-32}. Как следствие поиск коллизий стал возможен на современных суперкомпьютерах, что позволило создать два различных X.509 сертификата с одинаковой цифровой подписью, а также два различных документа в формате PostScript с одинаковой цифровой подписью [5].

31 марта 2005 г. Властимил Клима (Vlastimil Klima, Charles University in Prague) опубликовал улучшенный метод поиска коллизий, позволяющий выполнить необходимые расчеты на персональном компьютере [6].

Таким образом, в настоящий момент существует ряд достаточно эффективных методов поиска коллизий, что ставит под сомнение надежность цифровой подписи документов с использованием хэш-функции MD5.

В то же время неравенство множеств ключей и хэш-кодов позволяет выполнять как прямой перебор, так и обратный, по заранее рассчитанным хэш-таблицам (проект Rainbow), что позволяет определить исходный или эквивалентный ключ по хэш-коду в приемлемые сроки.

Цель статьи. В данной статье рассматриваются методы криптоанализа хэш-функции MD5 с целью раскрытия пароля и подделки цифровой подписи. Сделаны выводы относительно возможности и времени “взлома” хэш-кода MD5, приведены рекомендации по дальнейшему использованию данной хэш-функции, с учетом обнаруженных уязвимостей.

Основной раздел. Стандарт MD5 был разработан американским математиком Роном Ривестом (Ron Rivest). Впервые коллизию для MD5 удалось создать Ден Буру и Босселэру [1] для функции сжатия (т.е.

элементарной функции, которая на каждом раунде преобразует промежуточное значение, полученное из предыдущего раунда, и текущий блок сообщения в новое промежуточное значение) алгоритма MD5. Пока это не привело к компроментации MD5 в практических приложениях. Тем не менее, этот результат означает, что нарушен один из принципов построения MD5, а именно требование, чтобы функция сжатия была свободна от коллизий.

По своей природе проблема построения и использования коллизий алгоритма хэширования распадается на две задачи:

– криптографическая задача нахождения коллизии;

– практическая задача использования найденной коллизии для атаки на защищаемую алгоритмом хэширования систему.

Найденные подходы к построению коллизий являются значительным достижением в области криптографического анализа алгоритма MD5. Однако необходимо учитывать, что для успешного проведения целенаправленной атаки коллизии должны быть семантически допустимы. Используемые методы обнаружения коллизий не позволяют создавать коллизии с предопределенными значениями в заданных позициях. Поэтому угроза легитимности подписи на базе хэширования по алгоритму MD5 носит случайный характер для большей части документов. Подделка сертификата X.509 подобным способом позволяет модифицировать либо данные, либо ключ оригинала случайным образом с сохранением легитимности подписи. Однако в первом случае с большой вероятностью семантически значимые поля (имя, организация, дата выдачи и т.д.) будут испорчены произвольными символами, во втором случае будет создан произвольный открытый ключ, для которого невозможно будет создать закрытую половину. Таким образом, сертификат будет неработоспособен.

В то же время для документов сложной структуры, таких как RTF, XML, PostScript, изменение незначительного участка кода разметки может привести к замене блока текста скрытыми данными [5]. Злоумышленник, сумевший легально подписать особым образом сформированный документ (содержащий видимую и невидимую часть), получает возможность подменить значительную часть текста, используя коллизии для замены нескольких байт разметки документа.

В настоящий момент требуется оценка вероятности достижения положительного результата.

Таким образом, подделка или изменение документа без изменения цифровой подписи не может быть выполнена для документов простой структуры и сертификатов без их повреждения.

Документы сложной структуры рекомендуется перед подписью преобразовывать в формат, не допускающий прямого редактирования, например, сжимать архивирующими программами с контролем целостности.

При поиске ключа по хэш-коду методом прямого перебора, время поиска зависит от числа коллизий. Т.к. большему множеству ключей соответствует меньшее множество хэш-кодов, то перебор по меньшему множеству в настоящий момент является единственным способом увеличить скорость прямого перебора за счет использования коллизий, т.е. используются заранее обсчитанные, отсортированные таблицы (проект Rainbow) [7].

Размеры таблиц, время перебора и вероятность подбора на указанном множестве ключей для 7 символов [8]:

Таблица

Параметры таблиц Rainbow

Набор	*Длина*	*Размер*	*Время*	*Вероятность*
A-Z	7	610 Мб	31.1441 с	0.9990
A-Z, 0-9	7	3 Гб	40.6780 с	0.9904
A-Z, 0-9, @	7	18.3 Гб	275.3390 с	0.9990
Все	7	119 Гб	915.2542 с	0.9990

Наиболее простым способом устранения данной уязвимости является использование дополнительного алгоритма хеширования с объединением результатов для формирования хэш-кода. В этом случае множество кодов будет соразмерно множеству ключей и вероятность подбора эквивалентного ключа будет меньше в 2^x раз, где x – битовая длина кода дополнительного алгоритма.

Выводы. Несмотря на обнаруженные уязвимости, MD5 может быть и дальше использован для формирования цифровых подписей и шифрования паролей с незначительными изменениями.

Список литературы: 1. *Анохин М. И., Варновский Н. П. и др.* Криптография в банковском деле. – М.: МИФИ, 1997. – 300 с. **2.** *Молдовян А. А., Молдовян Н. А., Советов Б. Я.* Криптография. – М.: Лань, 2005. – 224 с. **3.** *Вильям Столлингс.* Криптография и защита сетей. Принципы и практика. – М.: Вильямс, 2001. – 672 с. **4.** *Xiaoyun Wang, Hongbo Yu*. How to Break MD5 and other Hash Functions. – Eurocrypto'05, 2005. **5.** *Stefan Lucks.* Attacking Hash Functions by Poisoned Messages: http://www.ruhr-uni-bochum.de/MD5Collisions.html. **6.** *Vlastimil Klima.* Finding MD5 Collisions on a Notebook PC Using Multi-message Modifications: **7.** *Нильс Фергюсон, Брюс Шнайер.* Практическая криптография. – М.: ISDN, 2005. – 424 с. **8.** *Исагулиев К. П.* Справочник по криптологии. – М.: Новое знание, 2004. – 238 с.

Поступила в редакцию 13.09.2005