УДК 004.056.55

Реализация однонаправленной хэш-функции на примере цифровой подписи MD2

Щетнева Валерия Андреевна
Приамурский государственный университет им. Шолом-Алейхема
студент

Глаголев Владимир Александрович
Приамурский государственный университет имени Шолом-Алейхема
к.г.н., доцент кафедры информационных систем, математики и методик обучения

Аннотация
В данной статье проведен анализ использование цифровых подписей для генерации шифротекса и разработана программа для реализации однонаправленной хэш-функции на основе цифровой подписи.
Ключевые слова: хэш-функция, цифровая подпись, шифрование, программа

Implementation of one-way hash functions for example digital signatures MD2

Shchetneva Valeria Andreevna
Sholom-Aleichem Priamursky State University
student

Glagolev Vladimir Aleksandrovich
Sholom-Aleichem Priamursky State University
candidate of geographical Sciences, associate Associate Professor of the Department of Information Systems, Mathematics and teaching methods

Abstract
In this article the analysis of use of digital signatures to generate cifroteca and developed a program to implement one-way hash functions based on digital signatures.
Keywords: hash function, digital signature, encryption program

Криптография сегодня - это важнейшая часть всех информационных систем: от электронной почты до сотовой связи, от доступа к сети Internet до электронной наличности. Криптография обеспечивает подотчетность, прозрачность, точность и конфиденциальность передаваемой информации. Она предотвращает попытки мошенничества в электронной коммерции и обеспечивает юридическую силу финансовых транзакций

Целью работы является разработка программы, реализующей однонаправленную хэш-функцию на примере цифровой подписи (сигнатуры).

Были поставлены следующие задачи: анализ литературы при разработке программы шифрования на основе однонаправленной хэш-функции; анализ различных алгоритмов создания цифровой сигнатуры; анализ назначения программы; разработка блок-схемы проектируемой программы.

Создание программы осуществлялось в среде программирование Delphi. В качестве данной функции использовался алгоритм MD2.

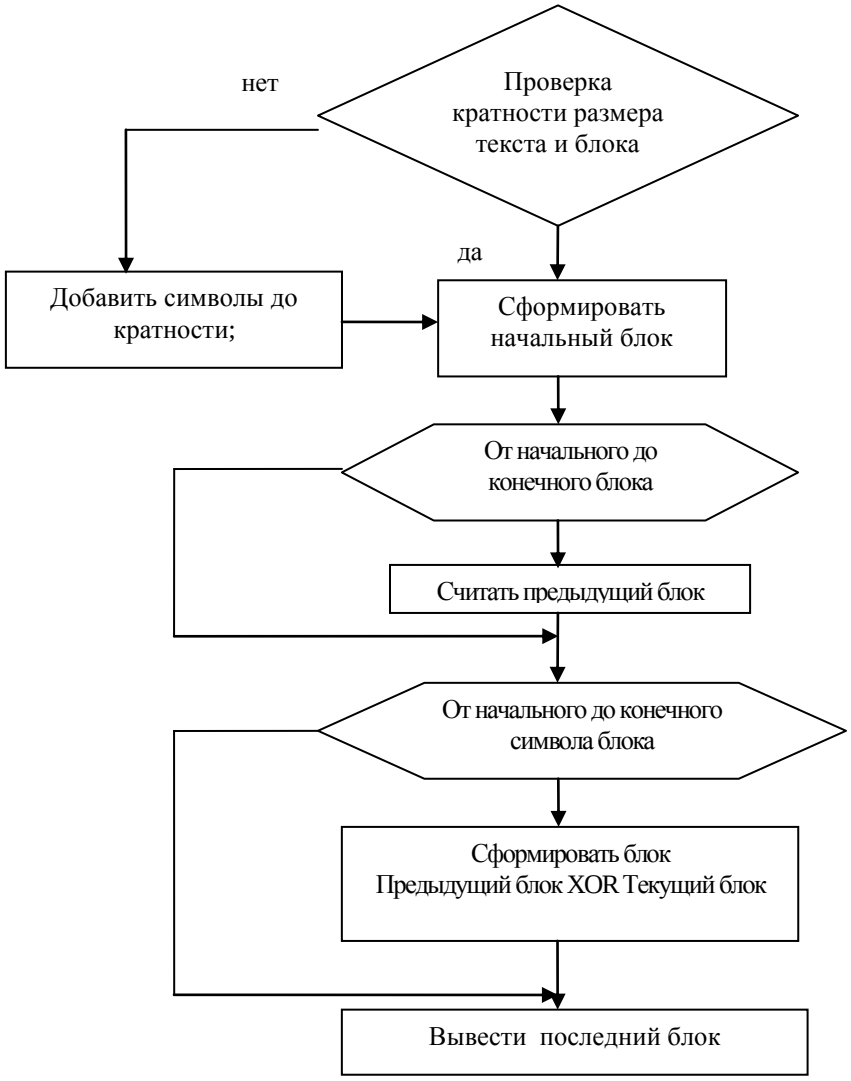

Рисунок 1 - Блок схема формирования цифровой сигнатуры

Разработанный алгоритм генерируют последовательностью символов, однозначно определяющих цифровую сигнатуры (см. рис. 1). Сигнатура создается определенным размером, количество байт разбиения на блоки. В результате выполнения программы формируется цифровая сигнатура добавляемая в конец текстового сообщения.

Опишем принцип, работы алгоритма с использованием блок-схемы.

Проверка кратности размера текста и блока – в зависимости от кратности, происходит добавление к текстовому сообщению необходимое количество символов.

Сформировать начальный блок – в зависимости от размер блока, формируется значение начального блока. Если размер блока 2 байта, то блок содержит значение «00».

Сформировать блок : = Предыдущий блок XOR Текущий блок – формирование значения следующего блока равно арифметической операции исключения ИЛИ между текущим и предыдущим.

Вывести последний блоки – выводим значение предыдущего блока, являющегося цифровой сигнатурой.

Чтобы хэшировать сообщения на основе цифровой подписи MD2 необходимо выполнить следующие этапы действий:

1. Дополнить сообщение i байтами, значение i должно быть таким, чтобы длина полученного сообщения была кратна n байтам.
2. Добавить к сообщению n байтов контрольной суммы.
3. Проинициализировать m-байтовый блок: X_o, X_1, X_2, . . ., Xm. Заполнить первые n байтов X нулями, во вторые n байтов X скопировать первые n байтов сообщения, а третьи n байтов X должны быть равны XOR первых и вторых n байтов X.
4. Функция сжатия, примерно выглядит следующим образом:

t:=0
For j = 0 to n+1
For k = 0 to m-1
$t:=X_t$ xor S_t
$X_k:=t$

5. Скопировать во вторые n байтов X вторые n байтов сообщения, а третьи n байтов *X* должны быть равны XOR первых и вторых n байтов *X*. Выполнить этап (4). Повторять этапы (5) и (4) по очереди для каждые n байтов сообщения.
6. Выходом являются первые n байтов *X*.

Для разработки программы, использовались следующие типы данных переменных:

- i,j:integer; - указатель символа текстового сообщения и блокаж
- res:array[0..100] of string; - массив текстовых блоков, размер блока определяется размером текстового сообщения;
- F1:TextFile; - файловая переменная текстового файла «output.txt»

После запуска программы необходимо указать размер блока цифровой подписи (см. рис. 2).

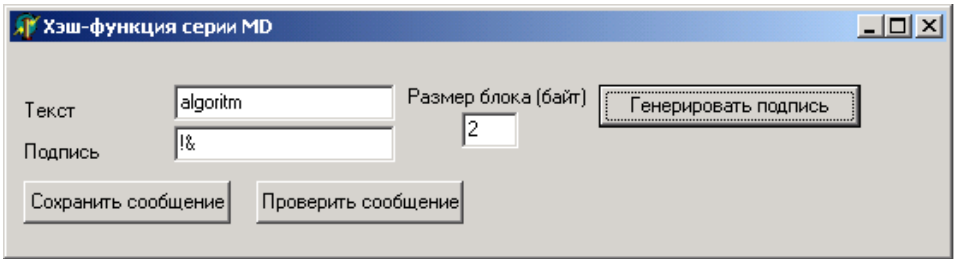

Рисунок 2 - Генерация цифровой подписи к текстовому сообщению

Генерация подписи просходит при помощи процедуры обработки нажатия на кнопку**.** В данном случае происходит запись текстового сообщения в переменную строковую переменную **r=itext.Text**, а затем определение кратности размера текстового сообщения и блока **if length(r) mod strtoint(rb.Text)<>0 then**. Если не кратно то программа происходит добавление символов '-' к тексту **for i:=1 to strtoint(rb.Text)-(length(r) mod strtoint(rb.Text)) do r:=r+'-';**Начальному блоку присваиваем значение от указанного размера **for i:=1 to strtoint(rb.Text) do res[0]:='0'+res[0];**

Формирование значения каждого блока равно арифметической операции исключения ИЛИ между текущим и предыдущим блоками посимвольно:

```
for i:=1 to round(length(r)/strtoint(rb.Text)) do
begin
s:=res[i-1];
p:='';
for j:=1 to strtoint(rb.Text) do
p:=chr(ord(s[j]) xor ord(r[j+(i-1)*strtoint(rb.Text)]))+p;
res[i]:=p;
```

Листинг 1. Арифметические операции алгоритма

При получение цифровой подписи для текстового сообщения «В результате при формирование является важным выбор размер, оптимальным является 4 байта, если длина сообщения больше 128 символов то 16 байт . т.к результат определяется на основе операции xor для определенного количества блоков (целая часть деления размера сообщения на размер блока).», результат при различных размерах подписи выглядит следующим образом (см. табл. 1).

Таблица 1 – Генерация цифровых подписей

Размер блока	Количество блоков	Подпись
3	95	цќw
4	72	WjЃ&
5	57	-^9
6	48	U±§%1?
7	41	РЏ^п-
8	37	Do`лэхг#
16	19	ЛЬЃ ффлх,?&.9/ръю

Таким образом, на основе анализа работы алгоритма MD2 была создана программа, приведен алгоритм выполнения программы при различных параметрах.

Библиографический список

1. Герасименко В.А. Защита информации в автоматизированных системах обработки данных. Кн. 1. М.: Энергоатомиздат. 1994. 400 с.
2. Вербицкий О.В. Вступление к криптологии. Львов.: Издательство науково-техничной литературы. 1998 300 с.
3. Баричев С. В. Криптография без секретов. М.: Наука, 1998. 120 с.