УДК 004.056.55

ЭВОЛЮЦИОННЫЕ МЕТОДЫ КОДИРОВАНИЯ ДАННЫХ, ПРИМЕР РАБОТЫ АЛГОРИТМОВ КОДИРОВАНИЯ И ДЕКОДИРОВАНИЯ

Н.И. КОРСУНОВ[1], А.И. ТИТОВ[1]
К.И. ЛОГАЧЕВ[2]

[1] Белгородский государственный национальный исследовательский университет

[2] Белгородский государственный технологический университет им. В.Г. Шухова

e-mail: titov@programist.ru
e-mail: korsunov@intbel.ru

В статье рассматривается работа алгоритма кодирования данных эволюционным методом с использованием ключа. Эволюционный процесс получения скрытного кода данных позволяет предотвратить выявление соответствий между открытым и скрытым кодом, методами частотного анализа.

Ключевые слова: скрещивание, ген, поколение, мутация, родитель, кодирование, декодирование, слово, ключ.

В данной статье для повышения криптостойкости кода предлагается использовать эволюционный подход, при котором шифр-код формируется в результате эволюционного отбора на заданном шаге селекции.

Алгоритм кодирования и декодирования рассмотрим на частном случае где четко заданно поколение на котором производится селекция, и не введена мутация генов. Реализация этого алгоритма подобна реализации алгоритма ГОСТ 28147-89 [1]. Что говорит о возможности описания всех существующих алгоритмов скрытного кодирования, как частные случаи эволюционного метода кодирования данных.

Здесь в качестве функции $x_1=f(x_0,y)$ может быть выбрана функция сложения по заданному модулю, как и в любой из известных систем скрытного кодирования данных[2].

Кодирование данных:

1) Смотрим в закрытый ключ первый и второй биты. В зависимости от значений закрепляем блоки данных «0» – закрепляем первый блок сова, «1»– закрепляем второй блок слова. Здесь первый бит информации отвечает за первое слово, второй бит за второе слово.

Оставшиеся блоки записываем из первого слова в третью позицию, из второго слова в четвёртую.

2) Делаем скрещивание C1 XOR C2 записываем результат в D1.

3) Смотрим в закрытый ключ третий бит, если:

«0» – берём C1 последний, C2 первый(записываем в D2);

«1» – берём C1 первый, C2 последний;

4) Если количество поколений не удовлетворяет критерию то повторить шаг 1-3, при этом ключ шифрования двигается дальше.

Наглядный пример работы алгоритма кодирования:

Ключ шифрования: 011100

Входные слова: первое слово(10001101)

второе слово(10110110)

Количество поколений для достижения стойкости: 2

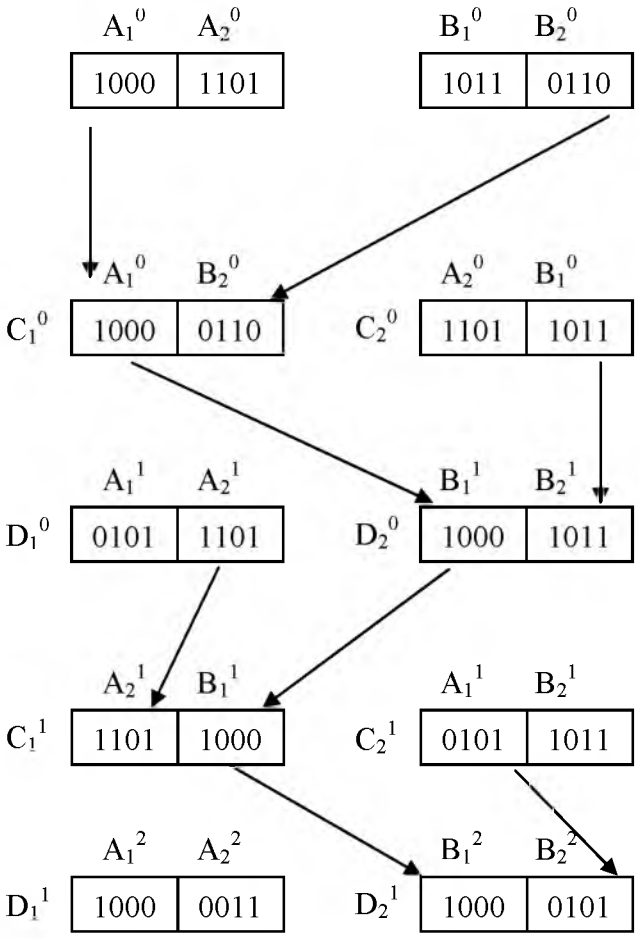

Рис 1. Работа алгоритма кодирования данных эволюционными методами

Декодирование данных предполагает наличие секретного ключа использованного при кодировании. Обмен ключом происходит по уже существующим закрытым каналам связи, или же во время «рукопожатия».

Декодирование:

1) Смотрим на ключ с конца.

Если бит имеет значение «0» то B_1^0 во второй блок слова $C1(Y_2)$, а B_2^0 в первый блок слова $C2(X_1)$;

Если бит имеет значение «1» то B_1^0 в первый блок слова $C1(Y_1)$, а B_2^0 во второй блок слова $C2(X_2)$;

2) В зависимости от значения считанного бита ключа на предыдущем шаге проводим восстановление предка по генотипу потомков.

Если «0» то $Y_1=(A_1^0 \text{ XOR } X_1)$; $X_2=(A_2^0 \text{ XOR } Y_2)$

Если «1» то $Y_2=(A_2^0 \text{ XOR } X_2)$; $X_1=(A_1^0 \text{ XOR } Y_1)$

3) Смотрим следующие с хвоста два бита ключа, в зависимости от их значений меняем блоки местами:

Таблица 1

Скрещивание блоков слова

Значение бит ключа	Запись блоков информации			
00	$A_1=Y_1$	$A_2=X_1$	$B_1=Y_2$	$B_2=X_2$
01	$A_1=Y_1$	$A_2=X_1$	$B_1=X_2$	$B_2=Y_2$
10	$A_1=X_1$	$A_2=Y_1$	$B_1=Y_2$	$B_2=X_2$
11	$A_1=X_1$	$A_2=Y_1$	$B_1=X_2$	$B_2=Y_2$

4) Если количество поколений не удовлетворяет критерию то повторить шаг 1-3, при этом ключ шифрования двигается дальше.

Наглядный пример работы алгоритма декодирования данных:
Ключ шифрования: 011100
Входные скрытые слова: первое слово(10000011)
второе слово(10000101)
Количество поколений для достижения стойкости: 2

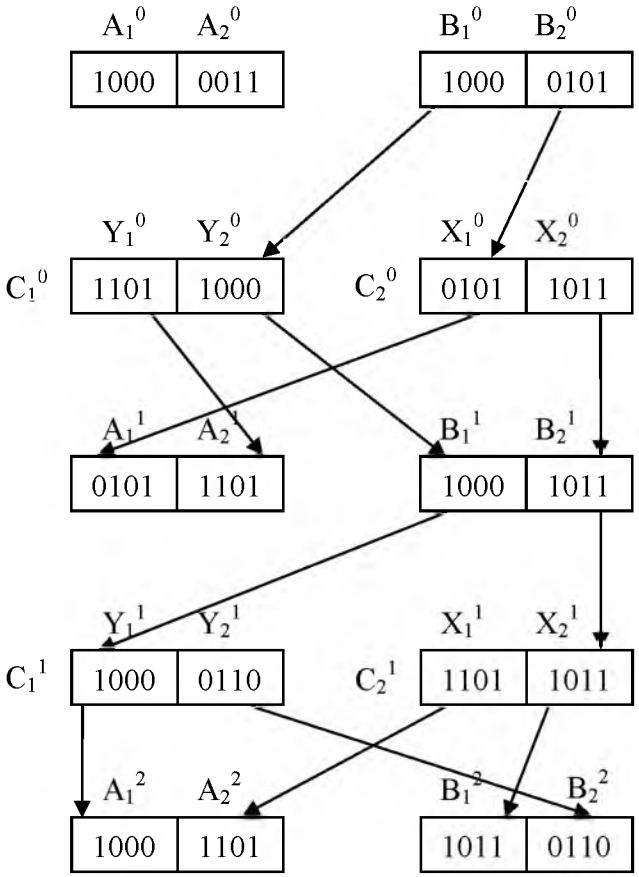

Рис 2. Работа алгоритма декодирования данных эволюционными методами

Для программной реализации не целесообразно использовать слова по 8 бит, так как работа с блоками по 4 бита предполагает резервирование ещё 4 бит, не используемых в программе[3]. Программный код для ускорения работы алгоритма необходимо реализовывать под работу со словами 16 бит, и блоками 8 бит, суть алгоритма кодирования не меняется так как не зависит от величины входного слова.

Так как при шифровании и дешифровании данных выполняются линейные операции, то сохраняется быстродействие, присущее методу шифрования перестановкой. Защищенность системы от взлома определяется скрытыми параметрами: количеством поколений N перед селекцией, закрытым ключом, сформированным случайным образом. В этом случае значение N позволяет осуществлять скрытное кодирование для отдельных абонентов.

Использование скрытого кодирования, зависящего от преобразуемых данных, имеет большое теоретическое и практическое значение. Теоретическая значимость заключается в обосновании нового класса примитивов и возможности расширения и совершенствования общих принципов построения итеративных схем блочных алгоритмов.

Теоретически важным является также, то обстоятельство, что математические свойства сложных, на первый взгляд, новых систем эволюционного кодирования данных[4], связанных с целой системой битовых преобразований, достаточно просто и эффективно определяются аналитически.

Работа выполнена при поддержке ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009-2013 годы, гос. контракт № 14.740.11.0390.

Список литературы

1. ГОСТ 28147-89. Системы обработки информации. Защита криптографическая. Алгоритм криптографического преобразования. — М.: Госстандарт СССР, 1989.

2. Зубов А.Ю. Совершенные шифры. – М.: Гелиос АРВ 2003 160 с., ил.

3. Молдовян, А. А. Криптография: скоростные шифры / А. А. Молдовян, Н. А. Молдовян, Н. Д. Гуц, Б. В. Изотов. – СПб.: БХВ-Петербург, 2002. – 496 с.

4. Корсунов Н.И., Муромцев В.В., Титов А.И. Метод расширения ключа для шифрования информации // Научные ведомости БелГУ. Серия: История, Политология, Экономика, Информатика. – 2010. – №19.

EVOLUTIONARY METHODS OF CODING DATA, AN EXAMPLE OF ENCODING AND DECODING ALGORITHMS

N.I. KORSUNOV[1], A.I. TITOV[1]
K.I. LOGACHEV[2]

[1] *Belgorod National Research University*

[2] *Belgorod Shukhov State Technological University*

e-mail: korsunov@intbel.ru
titov@programist.ru

The article examines the work of evolutionary algorithms encoding data by using the key. The evolutionary process of obtaining a concealed code data to prevent identification of correspondences between the open and hidden code, the methods of frequency analysis.

Key words: crossing, the gene, generation, mutation, arent, encoding, decoding, word key.