УДК 004.056

И.В. Лысенко

Национальный аэрокосмический университет имени Н.Е. Жуковского «ХАИ», Харьков

ПОДХОД К МОДИФИЦИРОВАНИЮ ПОДКЛЮЧЕЙ ДЛЯ КРИПТОАЛГОРИТМА ГОСТ 28147-89 В ЗАВИСИМОСТИ ОТ ПРЕОБРАЗУЕМЫХ ДАННЫХ

Предлагается подход к формированию расписания ключей криптоалгоритма ГОСТ 28147-89 в целях повышения его криптостойкости. В рамках данного подхода устраняется необходимость формирования сеансового ключа пользователей в каждом сеансе их взаимодействия.

Ключевые слова: *подключи, конфиденциальность, циклический сдвиг.*

Введение

В практике создания симметричных блочных криптоалгоритмов весьма редко встречаются алгоритмы шифрования, которые используют ключ шифрования (или его фрагменты) в «чистом» виде. Примером такого криптоалгоритма является стандарт шифрования ГОСТ 28147-89 (далее ГОСТ). Подавляющее большинство алгоритмов шифрования выполняет существенную модификацию исходного ключа для его последующего использования в процессе криптопреобразований.

Такая модификация называется расширением ключа или расписанием ключей (key extension, keys schedule) [1, 2].

На практике обычно формируется некоторое базовое отображение E', называемое раундовым, или цикловым, которое выполняется заданное число раз так, что в каждом раунде используется разные раундовые ключи $K^{(i)}$ (подключи), формируемые на основе базового ключа К. Объединение раундовых ключей называют расширенным ключом. Расширенный ключ можно представить в следующем виде [1]:

$$Q^{(e)} = Q^{(e,1)} \| Q^{(e,2)} \| \ldots \| Q^{(e,r)},$$

где e – режим преобразования данных (e = 0 – прямое и e = 1 – обратное преобразование данных) и $\forall j = 1,2,...,r\ Q^{(e,j)} = K^{(j+e(r-2j+1))}$. То есть ключ $Q^{(e)}$ является функцией от двух переменных, а именно:

$$Q^{(e)} = H(K, e),$$

где К – основной (базовый) ключ.

Процедура расширения ключа в алгоритме ГОСТ 28147-89 фактически отсутствует: в раундах шифрования последовательно используются 32-битные фрагменты K1…K8 исходного 256-битного ключа шифрования в следующем порядке: K1, K2, K3, K4, K5, K6, K7, K8, за исключением последних 8 раундов – в раундах с 25-го по 31-й фрагменты используются в обратном порядке.

Недостатком такого подхода к формированию раундовых ключей является то, что раундовые ключи являются явно зависимыми, что может быть использовано при криптоанализе. Достоинством непосредственного использования частей секретного ключа в качестве раундовых ключей является то, что обеспечивается сохранение высокой скорости шифрования в режиме частой смены ключей [3].

В этой связи в работе [4] рассмотрен подход к формированию расписания ключей криптоалгоритма ГОСТ, в основу которого положена идея случайного выбора элементов множества подключей на каждом раунде криптопреобразований. Это достигается за счёт формирования процедуры, которая по некоторому несекретному правилу в зависимости от начального ключа (всех 256 битов) позволяет каждому раунду криптопреобразования ставить в соответствие подключ K_j из заданного множества подключей (j = 1,…,8).

Ещё одним способом преодоления недостатка прямого использования подключей (использования ключа К в «чистом» виде) в раундах криптопреобразования является модифицирование подключей в зависимости от исходных (шифруемых) данных. В этом случае различные входные документы будут шифроваться с использованием различных наборов модифицированных значений подключей, что затрудняет вычисление последних [2].

В этой связи **целью статьи** является разработка модели формирования расписания ключей для блочного симметричного криптоалгоритма ГОСТ в зависимости от преобразуемых данных.

Модель формирования расписания ключа в зависимости от преобразуемых данных

Введём следующие обозначения:

– || – оператор конкатенации;

– >> (<<) – оператор циклического битового сдвига вправо (влево).

С учётом данных обозначений идея формирования (модификации) подключей формально может

быть представлена как:

$M_i = M_{Ri} \,||\, M_{Li}$,

$M_{Ri(2)} \rightarrow M_{Ri(10)}$, $M_{Ri(10)} (\mathrm{mod}\ 32) = S_j^i$ (j = 1,3,5,7),

F_1: $K_j \rightarrow K_j >> S_j^i \ \forall j = 1,3,5,7$;

$M_{Li(2)} \rightarrow M_{Li(10)}$, $M_{Li(10)} (\mathrm{mod}\ 32) = S_h^i$ (j = 2,4,6,8),

F_2: $K_j \rightarrow K_j << S_j^i \ \forall h = 2,4,6,8$.

Здесь M_{Ri} , M_{Li} – правая и левая части блока шифруемых данных M_i (i = 1,…,n), ∀i = 1,…,n: dim M_{Ri} = M_{Li} = 32 бита; $M_{Ri(2)}$, $M_{Ri(10)}$, $M_{Li\ (2)}$, $M_{Li\ (10)}$ – двоичное и десятичное представление правой и левой части блока шифруемых данных; S_j^i, S_h^i – параметр циклического сдвига вправо и влево соответственно (S_j^i,S_h^i ∈ {0, …, 31}); F_1, F_2 – отображения, задающие циклический сдвиг битов ключа на величину S_j^i и S_h^i соответственно.

Таким образом, согласно данной модели, биты всех подключей с *нечётными* номерами сдвигаются на величину, равную остатку от деления десятичного представления *правой* части блока шифруемых данных на 32, а биты всех подключей с *чётными* номерами сдвигаются на величину, равную остатку от деления десятичного представления *левой* части блока шифруемых данных на 32. При этом, как замечается в [4, с. 55], «если фиксированная операция циклического сдвига как частный случай операции перестановки является линейной операцией, то задание её зависимости от преобразуемых данных приводит к построению новой нелинейной операции с хорошими криптографическими свойствами».

Реализация данной модели может быть осуществлена в рамках гибридной криптосистемы: исходный ключ шифрования К длиной 256 битов надёжным способом передаётся от одного из пользователей другому, а в каждом сеансе взаимодействия пользователей шифруется n пар параметров циклического сдвига (n – число блоков сообщения) посредством открытого ключа несимметричного криптоалгоритма и в виде криптограммы передаётся вместе с шифртекстом для корректного расшифрования шифртекста получателем. Очевидно, что в данном случае по сравнению со стандартным подходом к использованию алгоритма ГОСТ увеличивается общий размер передаваемых зашифрованных данных на 10n битов (передаётся 2n параметров циклического сдвига, для представления каждого из которых используется 5 битов).

Заключение

Использование рассмотренного подхода, как предполагается, должно усилить криптостойкость алгоритма ГОСТ за счёт регулярной смены раундовых ключей в каждом сеансе, что также устраняет необходимость формирования сеансового ключа пользователей в каждом сеансе их взаимодействия, т.е. исходный ключ можно рассматривать как долговременный.

Целью дальнейших исследований является программная реализация предложенного подхода к формированию процедуры расширения ключа для алгоритма ГОСТ, а также проверка его работоспособности.

Список литературы

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

2. Молдовян А.А. Криптография / А.А. Молдовян, Н.А. Молдовян, Б.Я. Советов. – СПб.: Изд. «Лань», 2001. – 224 с.

3. Молдовян Н.А. Криптография. От примитивов к синтезу алгоритмов / Н.А. Молдовян, А.А. Молдовян, М.А. Еремеев. – СПб.: БХВ-Петербург, 2004. – 448 с.

4. Асташкина Е.Н. Подход к формированию расписания ключей для блочного симметричного криптоалгоритма ГОСТ 28147-89 / Е.Н. Асташкина, И.В. Лысенко // Системи обробки інформації. – Х.: ХУ ПС, 2010. – Вип. 6(87). – С. 30-34.

Поступила в редколлегию 13.12.2012

Рецензент: д-р техн. наук, проф. В.С. Харченко, Национальный аэрокосмический университет им. Н.Е. Жуковского «ХАИ», Харьков.

ПІДХІД ДО МОДИФІКАЦІЇ ПІДКЛЮЧІВ ДЛЯ КРИПТОАЛГОРИТМА ГОСТ 28147-89 В ЗАЛЕЖНОСТІ ВІД ДАНИХ, ЩО ПЕРЕТВОРЮЮТЬСЯ

І.В. Лисенко

Пропонується підхід до формування розкладу ключів криптоалгоритма ГОСТ 28147-89 з метою підвищення його криптостійкості. В межах даного підходу усувається необхідність у формуванні сеансового ключа користувачів в кожному сеансі їх взаємодії.

Ключові слова: *підключи, конфіденційність, циклічний зсув.*

THE APPROACH TO THE MODIFICATION OF SUBKEYS FOR CRYPTOALGORITHM GOST 28147-89 IN DEPENDENCE OF TRANSFORMED DATA

I.V. Lysenko

In order to increase cryptographic proofness of the cryptographic algorithm GOST 28147-89 an approach to the formation of keys schedule for this algorithm is proposed. This approach eliminates the need for a user session key for each session of their interaction.

Keywords: *subkeys, confidentiality, cyclic shift.*