УДК 621.327

АЛГОРИТМ ГИБРИДНОГО ШИФРОВАНИЯ

А.А. Смирнов, В.Е. Чевардин
(представил д.т.н., проф. Ю.В. Стасев)

В статье предлагается метод шифрования, основанный на гибриде симметричных и асиметричных схем преобразований информации. Расмотрены временные характеристики операций, выполняющихся при криптографических преобразованиях информации.

Тенденции развития современных компьютерных технологий требуют глубокого анализа существующих методов криптографической защиты в компьютерных системах и сетях. При программной или программно-аппаратной реализации различных криптографических методов наиболее ответственной операцией является чтение секретных ключей и паролей. Поэтому одной из проблем на сегодня является анализ набора и порядка операций в различных алгоритмах шифрования и дешифрования информации.

Основными функциями в существующих алгоритмах шифрования являются следующие операции: возведение в **n** - ую степень, сложение по модулю, умножение, циклический сдвиг. Анализ уязвимости различных операций [1] с точки зрения временных характеристик представлен в табл. 1.

Таблица 1

Уязвимость криптографических операций от временных атак

Операция	Уязвимость для временных атак
Поиск по таблицам (для шифров с заменой)	Неуязвим для временных атак
Фиксированные сдвиги	Неуязвимы для временных атак
Булевы операции	Неуязвимы для временных атак
Сложение/вычитание	Трудно защитить от временных атак
Умножение/деление	Наиболее уязвимы для временных атак

Временные затраты определяются количеством тактов, необходимых для выполнения соответствующих криптографических операций. Зная временные параметры используемых операций, можно произвести криптографическую атаку. Количество тактов, необходимое для выполнения различных операций [2], приведено в табл. 2, где **m, n, k** – количество 32 - битовых блоков в числах $\mathbf{U = (U_1 U_2 \ldots U_m)}$, $\mathbf{V = (V_1 V_2 \ldots V_n)}$, $\mathbf{P = (P_1 P_2 \ldots P_k)}$.

Видно, что криптоаналитик, имея хотя бы косвенный доступ к компьютерной системе (с помощью программного модуля - скрипта), помимо опре-

деления операций, выполняющихся компьютерной системой, может определить временные параметры между операциями, тем самым приблизится к распознаванию алгоритма шифрования.

Таблица 2

Количество тактов, необходимое для выполнения различных операций

Операция	Математическое представление	Количество тактов, необходимое для выполнения операции
Умножение по модулю	$\mathbf{V \cdot U \ (mod\ P)}$	$\mathbf{982 \cdot n^2 + 527{,}5 \cdot n + 347}$
Возведение в степень по модулю	$\mathbf{U^V \ (mod\ P)}$	$\mathbf{8736 \cdot n^3 + 2726 \cdot n^2 + 18862 \cdot n}$
Сложение	$\mathbf{U+V}$	$\mathbf{28 \cdot n + 26}$
Вычитание	$\mathbf{U-V}$	$\mathbf{28 \cdot n + 26}$
Умножение 1	$\mathbf{U \cdot V}$	$\mathbf{100 \cdot m \cdot n + 52{,}5 \cdot m + 8{,}5 \cdot n + 122}$
Умножение 2	$\mathbf{U \cdot V}$	$\mathbf{75 \cdot n^2 + 287{,}5 \cdot n + 548}$
Деление	$\mathbf{U/V}$	$\mathbf{82 \cdot (m-n) \cdot m + 276 \cdot (m-n) + 126 \cdot m + 42 \cdot n + 228}$

Один из наиболее стойких на сегодня - алгоритм RSA [1]. Он имеет некоторые недостатки в применении: это, во-первых, низкое быстродействие (время выполнения достаточно велико - на сегодняшний день ни одна из криптосистем с открытым ключом не может конкурировать по этому показателю с криптосистемами с секретным ключом), а во-вторых, резкое отличие в работе системы по этому алгоритму от других алгоритмов - узнаваемость алгоритма, что в какой-то степени облегчает компрометацию.

Основной операцией в RSA алгоритме является процедура возведения в степень по модулю, где операнды и модуль представляют собой целые числа большой разрядности. В основе возведения в степень положен бинарный метод [3] (метод возведения в квадрат и умножения), предполагающий:

- последовательное вычисление $\mathbf{X^2, X^4, \ldots, X^{2^{m_e-1}}}$,

где $\mathbf{m_e}$ - число разрядов ключа, **X** - зашифрованый текст;

- перемножение степеней, необходимых для набора степени, в которую возводят.

Время выполнения схемы умножения оценивается формулой:

$$\mathbf{T_y = (2m_e - 2) \cdot t_{mu1} + m_e \cdot t_k} \qquad (1)$$

где $\mathbf{t_{mu1}}$ - время одного умножения по модулю **n**; $\mathbf{t_k}$ - время работы одного коммутатора (управляющего оператора).

Способы построения умножителей по модулю **n** нетривиальны, их выбор зависит от требований к быстродействию определяемыми требованиями к скорости процедуры шифрования.

Если требуемая скорость процедуры шифрования равна **V** (бит/сек), то требуемое время шифрования одного сообщения равно

$$\mathbf{T_{тр}} = \frac{\mathbf{m_x}}{\mathbf{V}}, \qquad (2)$$

где $\mathbf{m_x}$ - разрядность зашифрованого сообщения.

В конвейерном режиме принимается следующий друг за другом текст последовательно с периодом $\mathbf{T_k \geq \tau_x}$, где $\boldsymbol{\tau_x}$ - время выполнения одного умножения.

Для организации конвейерного режима необходимо осуществлять синхронизацию между отдельными операциями умножения (синхронизацию умножителей по модулю).

Тогда требуемое время умножения по модулю будет равно:

1) при паралельном режиме работы

$$\mathbf{T_{тр\,1}^{умн}} = \frac{\mathbf{T_{тр}}}{\mathbf{m_e}} = \frac{\mathbf{m_x}}{\mathbf{V \cdot m_e}}; \qquad (3)$$

2) при конвейерном режиме работы

$$\mathbf{T_{тр\,2}^{умн}} = \mathbf{T_{тр}} = \frac{\mathbf{m_x}}{\mathbf{V}}. \qquad (4)$$

Отсюда следует, что при конвейерном режиме схемы шифрования повышение скорости может быть достигнуто, если каждая операция умножения будет выполняться отдельной ЭВМ, в которой умножение реализуется программой. В этом случае преобразователь будет представлять собой по существу многомашинный вычислительный комплекс.

Если же конвейерный вариант неприемлем, то в связи с более жесткими требованиями к быстродействию, умножитель необходимо реализовывать аппаратно, либо на основе специального программного обеспечения с использованием частичного преобразования сомножителей. Однако, при любом варианте нам могут быть известны временные характеристики выполнения операций, использующихся при реализации RSA алгоритма.

Таким образом, во избежание узнаваемости алгоритма шифрования, снижения временных затрат на шифрование информации предлагается использовать схемы гибридного шифрования, одна из которых представлена на рис. 1.

Исходя из анализа уязвимости к временным атакам следует, что секретный ключ $\mathbf{k_{AB} = \omega^{X_A X_B}}$ надо формировать по схеме RSA, а для шифрования массива информации применять замену по таблицам, которые формировать, в свою очередь, с помощью генератора псевдослучайных последовательностей, использующего секретный ключ $\mathbf{k_{AB} = \omega^{X_A X_B}}$. Помимо этого, предлагается определенные блоки массива информации шифровать по RSA - схеме.

Реализацией предлагаемого алгоритма нам удастся решить вопрос о

быстродействии алгоритма шифрования, передачи ключа по открытому каналу связи, но не в явном виде, а самое главное, уменьшить узнаваемость алгоритма шифрования. Однако для оптимального выполнения

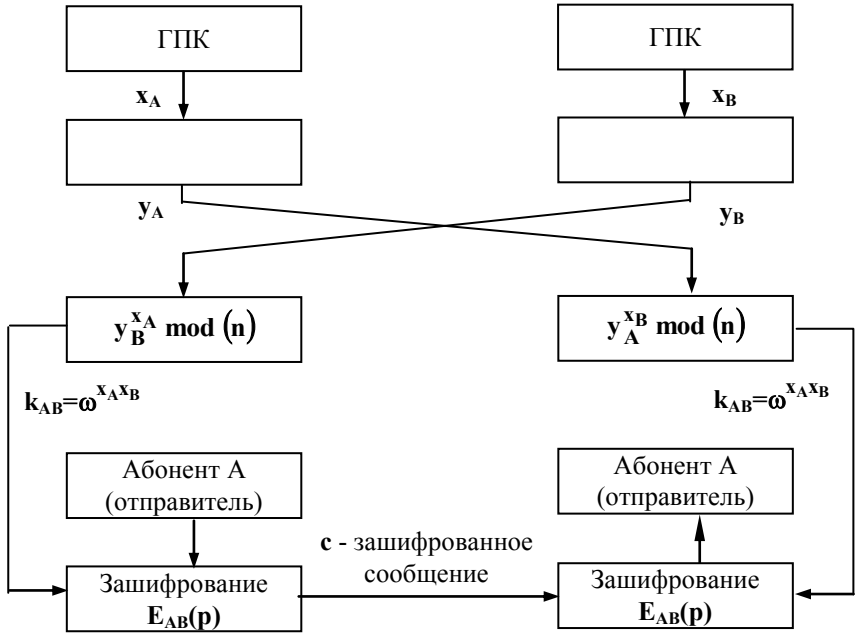

Рис. 1. Вариант схемы гибридного шифрования

вышепредложенного алгоритма необходимо предъявлять жесткие временные требования к каждому этапу криптопреобразований.

ЛИТЕРАТУРА

1. *Иванов М.А. Криптографические методы защиты информации в компьютерных системах и сетях. – С - Пб: Питер, 2000. – 354 с.*
2. *Кнут Д. Искусство программирования для ЭВМ. Т.1. Основные алгоритмы. – М.: Мир, 1976. – 735 с.*
3. *Кнут Д. Искусство программирования для ЭВМ. Т.2. Получисленные алгоритмы. – М.: Мир, 1977. – 724 с.*

Поступила 02.04.2002

***СМИРНОВ Алексей Анатольевич**, адъюнкт ХВУ. В 1999 году окончил ХВУ. Область научных интересов – защита информации в автоматизированных системах управления и сетях.*

***ЧЕВАРДИН Владислав Евгеньевич,** в 2001 году окончил ХВУ. Область научных интересов – защита информации в автоматизированных системах управления и сетях.*