В.А. Козлов

Программный комплекс симметричных криптографических преобразований с использованием методов биграммной шифрации

Криптографический способ защиты информации предусматривает такое ее преобразование, при котором она становится доступной для прочтения лишь обладателям некоторого секретного параметра – ключа.

Современные криптосистемы подразделяются на асимметричные системы с открытым ключом и симметричные с секретным ключом.

Для алгоритмов асимметричного шифрования характерно применение нелинейных аналитических преобразований шифруемого текста. Асимметричное шифрование характеризуется следующим соотношением ключей: $K_1 = f(K_2)$, где K_1 – открытый ключ; K_2 – секретный ключ. *Открытый ключ асимметричного шифрования* – ключ, с помощью которого производится зашифрование информации. *Секретный ключ асимметричного шифрования* – ключ, с помощью которого производится расшифрование информации.

Основным пунктом концепции асимметричного шифрования является предложение использовать ключи парами, состоящими из ключа зашифрования и ключа расшифрования, которые невозможно вычислить один из другого [1]. Основными недостатками алгоритмов асимметричного шифрования являются:

– низкая скорость выполнения операций зашифрования и расшифрования;

– отсутствие математически доказанной криптостойкости алгоритмов асимметричного шифрования;

– необходимость защиты открытых ключей от подмены.

Симметричные криптосистемы построены на основе сохранения в тайне ключа шифрования. Процессы зашифрования и расшифрования используют один и тот алгоритм и один и тот же ключ. Секретность ключа является постулатом. Основная проблема при применении симметричных криптосистем для связи заключается в сложности передачи обоим сторонам секретного ключа. Алгоритмы симметричного шифрования базируются на применении в различных сочетаниях всего двух принципов: *рассеивания* и *перемешивания* [2].

Рассеивание представляет собой распространение влияния каждого знака открытого текста на каждый знак шифртекста, что позволяет скрыть статистические свойства открытого текста.

Перемешивание предполагает использование таких шифрующих

преобразований, которые усложняют восстановление взаимосвязи статистических свойств.

Стойкий шифр с хорошим рассеиванием и перемешиванием получается при многократном чередовании простых *перестановок* и *замен*, управляемых достаточно длинным секретным ключом.

Именно такой подход использовался практически во всех современных алгоритмах симметричного шифрования. Алгоритм составного шифрования состоит из трех этапов. На предварительном этапе информация разбивается на блоки одинаковой длины и производится ее начальная перестановка по какому-либо закону. В рамках второго этапа с помощью ключа шифрования выполняется основное криптографическое преобразование. Полученный промежуточный результат модифицируется с использованием алгоритмов табличной замены. Далее основной криптографический цикл повторяется еще несколько раз в зависимости от конкретного алгоритма. На третьем этапе выполняется финальная перестановка, ее результатом является блок зашифрованной информации [3].

В работе [4] приведено описание оригинального алгоритма биграммной подстановки «двойная пирамида», который использует две совмещенные по горизонтали пирамиды (рис. 1). Количество строк у параллелограмма, объединяющего эти две пирамиды – 16. Длина каждой объединенной строки – 32 ячейки. Таким образом, объединенная из двух пирамид фигура включает в себя $16 \cdot 32 = 512$ ячеек, следовательно, каждая пирамида состоит из 256 ячеек.

Процесс шифрации осуществляется следующим образом:

– выбираем очередную пару символов исходного текста и находим в левой пирамиде ячейки с соответствующими кодами;

– в правой пирамиде по правилам 1-3 находим две ячейки, коды которых становятся биграммной парой шифра текста (более подробное описание алгоритма приведено в работе [5]).

В настоящее время завершена разработка программного комплекса по реализации вышеприведенного алгоритма, реализованная в нескольких режимах функционирования: инициализации, демонстрационный, рабочий и комплексный. Программный комплекс обеспечивает шифрацию и дешифрацию файлов и передачу их по сети. Инсталляция не требуется, программа представляет собой .exe-файл небольшого размера. Для работы программы в комплексном режиме необходимо дополнительно установить на Вашем компьютере бесплатную программу *Pretty Good Privacy* (PGP). Каких-либо строгих ограничений по производительности используемого компьютера нет. Программный комплекс реализован на языке Delphi в среде Borland Developer Studio 2006.

Интерфейс режима инициализации. На рис. 2 приведена головная форма, с помощью которой осуществляется выбор режима функционирования программы. Настройка программных модулей осуществляется в режиме инициализации (кнопка «Инициализация»). Режим инициализации включает в себя два варианта настройки: структурную и информационную.

Структурная настройка позволяет пользователю осуществить осознанный выбор тех элементарных алгоритмов преобразования, которые войдут в состав блока основных преобразований. Выбор нужных алгоритмов производится пользователем из соответствующего списка.

После выбора перечня используемых алгоритмов криптографических преобразований следует перейти к информационной настройке, которая заключается в заполнении исходников случайными неповторяющимися значениями от 0 до 255 [5]. Количество исходников определяется либо пользователем, либо в автоматическом режиме по соответствующему алгоритму. По умолчанию количество исходников равно 256. Содержимое исходников не является секретной информацией, так как на их основе создаются рабочие файлы, которые используют секретную ключевую комбинацию, маркант и алгоритм гаммирования для создания секретных рабочих файлов той же размерности.

По умолчанию исходники в количестве 256 штук поставляются в комплекте с программой. Таким образом, исходники представляют собой открытую нормативно-справочную информацию, которую, однако, по договоренности сторон можно изменить в любой момент времени.

Единственным, но неукоснительно выполняемым условием должно быть то, что исходники обменивающихся информацией сторон должны полностью совпадать. В противном случае процессы шифрации и дешифрации окажутся некорректными для каждого из партнеров.

Интерфейс демонстрационного режима. В процессе разработки демонстрационного режима преследовались следующие цели:

– упростить процесс обучения неподготовленного пользователя;

– убедить пользователя в криптостойкости используемых алгоритмов;

– показать степень оперативности реализации каждого алгоритма в отдельности и программы в целом.

Меню головной формы предоставляет пользователю возможность выбора для реализации как отдельных алгоритмов криптографических преобразований, так и всей программы в целом.

Демонстрационный режим обеспечивает возможность в динамике просмотреть все процессы и алгоритмические преобразования, кото-

рые реализуются по ходу выполнения математических операций. Последнее обеспечивается тем, что все формы этого режима имеют специальное встроенное окно «*Мониторинг работы алгоритма*», в котором отражаются все элементарные операции над каждым символом исходного текста по ходу выполнения того или иного алгоритма криптографических преобразований.

Рассмотрим некоторые из них. Настройка программных модулей на этапе инициализации нами уже рассматривалась в предыдущем разделе, при этом для наглядности описания процессов функционирования отдельных алгоритмов в режиме инициализации был выбран интерфейс именно демонстрационного режима.

А*лгоритм перемешивания.* Подробное описание алгоритма приведено в работе [5]. Для его реализации необходимо ввести в окно «*Парольная фраза*» ключевую секретную комбинацию, с помощью которой, по одному из исходников будет сгенерирована рабочая таблица, посредством которой каждый символ шифруемого текста займет свое место.

В рамках демонстрационного режима предполагается, что длина каждого блока шифруемого текста не превышает 255 символов (первый символ служебный, он указывает, какое количество значащих символов содержит шифруемый текст).

При этом в окне «*Загруженный ключ шифрования*» появляется содержимое выбранного исходника, посредством которого будет в дальнейшем сформирована рабочая таблица. Номер выбранного исходника из общего множества существующих появляется в окне «*Мониторинг работы алгоритма*».

После того как выбранный исходник загружен, можно приступать к формированию рабочей таблицы. С этой целью необходимо активизировать кнопку «*Сформировать рабочую таблицу*». Вид формы после нажатия кнопки представлен на рис. 9. Значение элементов таблицы появится в окне «*Сформированная рабочая таблица*». В окне «*Мониторинг работы алгоритма*» представлен полный перечень итераций по перемещению элементов исходника в процессе генерации рабочей таблицы. Теперь, когда предварительный этап завершен, можно приступать к процессу шифрации исходного текста или дешифрации шифрограммы.

С этой целью необходимо либо набрать исходный текст в окне «Текст для шифрования», и нажать на кнопку «*Зашифровать текст*», либо активировать кнопку «*Файлы*►*Зашифровать*», осуществить поиск нужного файла и открыть его.

В первом случае зашифрованный текст появится в окне «*Зашифрованный текст*», а в окне «*Мониторинг работы алгоритма*» для каж-

дого элемента сформированного шифртекста будет сгенерирован либо его исходный номер, либо зафиксирован факт того, что это ничего не значащий случайный символ. Во втором случае зашифрованный файл копируется в папку программного каталога «*Зашифровано*» в каталоге запуска программы с тем же именем и с тем же расширением.

Алгоритм «Двойная пирамида». Подробное описание алгоритма приведено в работе [4]. Загружаем демонстрационный режим, вызываем алгоритм «Двойная пирамида», после чего появляется соответствующая форма.

После занесения кодовой комбинации в окно «*Парольная фраза*» и нажатия на кнопку «*Ключ шифрования ► Загрузить*» в окне «*Загруженный ключ шифрования*» появляется поэлементно заполненный матричный эквивалент «двойной пирамиды». Далее заносим исходный текст в многострочный редактор «*Текст для шифрования*» и нажимаем на кнопку «*Зашифровать*» в окне «*Шифрование*» и получаем шифр, который появляется в окне «*Зашифрованный текст*».

Если исходный текст помещен в каком-либо файле, то с помощью окна «*Файлы*» и кнопки «*Зашифровать*» отыскиваем по каталогу файл, подлежащий шифрованию, открываем его и нажимаем на кнопку «*Зашифровать*». Зашифрованный файл с тем же именем помещается в программный каталог «*Зашифровано*».

Комплексный алгоритм симметричных криптографических преобразований. Демонстрационный режим работы алгоритма осуществляет демонстрацию работы всей программы в целом.

Интерфейс рабочего режима. В рабочем режиме осуществляется шифрация и дешифрация исходных сообщений и различного рода файлов. Головная, да и другие формы основного режима идентичны формам демонстрационного режима с той лишь разницей, что в них отсутствуют такие окна, как «*Мониторинг работы алгоритма*», «*Загруженный ключ шифрования*», «*Сформированная рабочая таблица*» и т.д.

Библиографический список

1. Брюс Шнайдер Прикладная криптография. М.: ТРИУМФ, 2002. 816 с.
2. Масленников М.Е. Практическая криптография. СПб.: БХВ-Петербург, 2003. 464 с.
3. Краковский Ю.М. Информационная безопасность и защита информации: учеб. пособие. М.: ИКЦ «МарТ», 2008. 288 с.
4. Козлов В.А. и др. Алгоритм криптографических преобразований на базе биграммного шифра «двойная пирамида». Университетские чтения – 2010. Материалы научно-методических чтений ПГЛУ. Часть XV. Пятигорск: ПГЛУ, 2010. С. 141-148.
5. Козлов В.А. Алгоритм криптографических преобразований с использованием методов биграммной шифрации // Управление и информационные технологии. Межвузовский научный сборник. Пятигорск: РИА на КМВ, 2010. С. 121-132.