УДК 004.623

Безопасная компоновка криптографических модулей как способ защиты информации

В работе предлагается способ безопасной компоновки криптографических модулей для обеспечения защиты информации. Представлены функциональная схема, состав универсального криптографического модуля и описан процесс обработки входных/выходных параметров. Применение безопасной компоновки позволяет обеспечить надежную загрузку и выполнение программного модуля.

А. П. Мартынов, Д. Б. Николаев, М. В. Одинцов, Д. В. Сплюхин

Обеспечение защиты информации – одно из главных требований, предъявляемых к вычислительной технике, участвующей в процессах обработки, хранения и передачи конфиденциальной информации. Защищенность информации обеспечивается применением криптографических систем в совокупности с организационно-техническими мероприятиями. В последнее время при построении криптографических систем увеличивается роль программных средств защиты информации, просто модернизируемых и не требующих крупных финансовых затрат в сравнении с аналогичными аппаратными системами. Для решения задач по защите информации с использованием программных средств значительное внимание уделяется вопросам криптографической стойкости алгоритмов преобразования, гарантированного обеспечения целостности данных и возможности модульного программного применения на различных платформах.

Существует множество разнообразных криптографических алгоритмов преобразования информации, которые гарантируют высокую степень защищенности данных. Криптографические алгоритмы могут быть реализованы в виде криптографических программных модулей – программных устройств для преобразования информации по криптографическим алгоритмам, из которых формируются криптографические программные средства [1]. В идеальном случае все криптографические алгоритмы должны оформляться в виде независимых модулей для гарантирования дальнейшей работоспособности программных криптографических средств.

С учетом вышесказанного возникает задача безопасной компоновки программных модулей, реализующих различные криптографические алгоритмы и построенных по определенному принципу. Унификация и объединение криптографических программных модулей представляет собой безопасную загрузку программных модулей в контролируемую область памяти и исключает возможность их исполнений, способных привести к сбою программного средства в целом. При функционировании главной программы к ней динамически подключается программный модуль, функции модуля экспортируются в программу для выполнения требуемых действий. При этом главной программе передаются только параметры приема функций подключенного программного модуля и параметры обратного процесса передачи, а параметры преобразования внутри программы неизвестны. За счет этого исключается возможность несанкционированного доступа к функциям программных модулей.

Подключение программных модулей обеспечивается с помощью «библиотек динамической компоновки», которые могут быть многократно задействованы программными средствами. Способ получения адреса функции основан на смещении начала функции от начала «библиотеки динамической компоновки», которая является величиной постоянной, не зависящей от процесса. Данная процедура осуществляется следующим образом:

– создается новый процесс в адресном пространстве [2];

– происходит загрузка в новый созданный процесс «библиотеки динамической компоновки»;

– осуществляется получение адреса нужной функции «библиотеки динамической компоновки»;

– происходит декрементация из адреса функции адреса загрузки «библиотеки динамической компоновки»;

– осуществляется инкрементация к получившемуся смещению адреса загрузки «библиотеки динамической компоновки» в созданном процессе;

– на выходе выдается адрес функции в созданном процессе.

Очевидно, что если «библиотека динамической компоновки» в двух разных процессах загружена по одному адресу, то и адреса функций будут совпадать. А поскольку в нормальных, не слишком сложных процессах системные «библиотеки динамической компоновки» формируются по одним и тем же адресам, адреса системных функций во всех процессах одинаковы. Под безопасной компоновкой будем понимать процесс обеспечения загрузки программного модуля в контролируемую область памяти и исключение возможности действий программного модуля, способных привести к сбою программы в целом.

Программная реализация более практична, допускает известную гибкость в использовании и предоставляет возможность реализации различных криптографических преобразований [3]. Программный криптографический модуль состоит из экспортируемых функций, которые задаются инженерами или программистами. Нередко случается, что ошибки программистов и инженеров влияют на безопасность и надежность программирования. Возвращаемыми значениями при загрузке модуля являются: название криптографического алгоритма, формат ключевой информации, формат ввода информации из файла или размерность блока чтения из файла, описание криптографического алгоритма (руководство программиста).

Программный криптографический модуль выполняет следующие функции:

– создание и удаление ключевой информации;

– передача информации и названия криптографического алгоритма;

– преобразование информации по криптографическому алгоритму;

– обратное преобразование по криптографическому алгоритму.

Взаимодействие криптографического модуля с входными и выходными параметрами показано на рис. 1.

Для обеспечения безопасности программы от нелегального запуска можно преобразовать программный модуль, в этом случае программный продукт не сможет выполнить загрузку, загружаемый модуль будет защищен от заражения вирусом, дисассемблирования и программной отладки, даже копирование преобразованного файла не приведет к желаемому результату.

В основе безопасной компоновки лежит формирование всей программы или модуля программы в безопасном сегменте оперативной памяти, исключающее негативное (недекларируемое) воздействие данной программы (модуля) на другие программы (модули) на время выполнения программы. Применение безопасной компоновки позволяет в определенной степени обеспечить безопасное функционирование применяемых программных модулей.

Рис. 1. Взаимодействие криптографического модуля с входными и выходными параметрами

После подключения программного модуля осуществляется его проверка на целостность и правильность структуры. Структура программного модуля представлена на рис. 2.

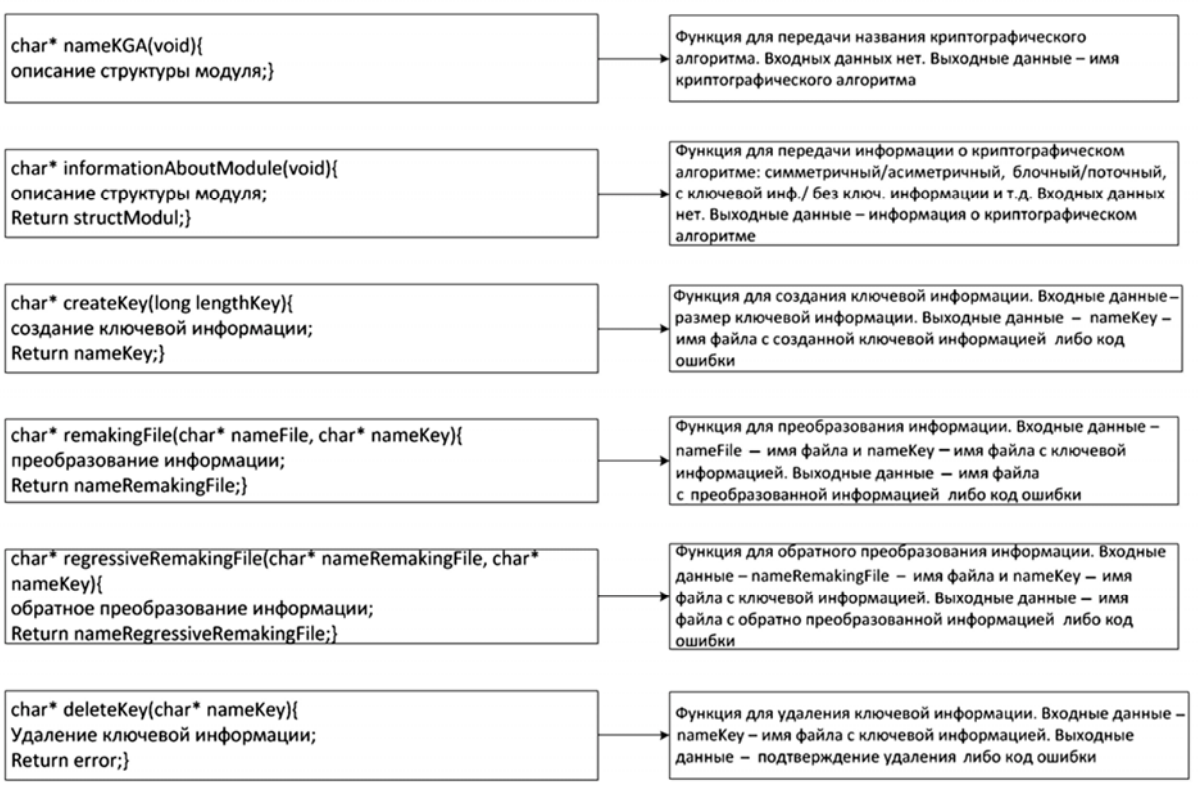

Рис. 2. Функциональная структура криптографического программного модуля

На следующем этапе запуска программы происходит чтение файла, в котором записаны файлы подключаемых программных криптографических модулей. Программа, после подключения программных модулей, хранится в оперативной памяти в преобразованном виде, и доступ к ней может получить практически любой пользователь системы. Процедура получения доступа к открытому исполняемому файлу затруднена, так как загрузка программы или программного модуля осуществляется только на момент выполнения в контролируемую область памяти. При

положительном результате аутентификации управление передается из основной программы функциям подключаемых модулей. В противном случае выполнение программы прекращается и выдается сообщение о недопустимом действии. По окончании сеанса работы программы вся занимаемая программой часть оперативной памяти должна очищаться.

Таким образом, при использовании безопасной компоновки обеспечивается комплексная защита выполняемых программных модулей в части проверки целостности данных и функциональной верификации алгоритмов.

Список литературы

1. Грибунин В. Г., Костюков В. Е., Мартынов А. П., Николаев Д. Б., Фомченко В. Н. Современные методы обеспечения безопасности информации в атомной энергетике. – Саров: ФГУП «РФЯЦ-ВНИИЭФ», 2014. 636 с.

2. Запонов Э. В., Мартынов А. А., Марунин М. В. Схемотехническое построение элементов электронно-вычислительных машин. – Саров: ФГУП «РФЯЦ-ВНИИЭФ», 2015.

3. Мартынов А. П., Николаев Д. Б., Новиков А. В., Фомченко В. Н. Промышленные интерфейсы для научных исследований. – Саров: ФГУП «РФЯЦ-ВНИИЭФ», 2013. 238 с.

Safe configuration of cryptographic modules as the way of protection of the information

A. P. Martynov, D. B. Nikolaev, M. V. Odintsov, D. V. Splyukhin

In work the way of safe configuration of cryptographic modules for maintenance of protection of the information is offered. The function chart, structure of the universal cryptographic module are presented and process of processing of entrance / target parametres is described. Application of safe configuration allows to provide reliable loading and performance of the program module.