ЗАЩИТА ПРОГРАММНОГО КОДА ОТ НЕСАНКЦИОНИРОВАННОГО ИЗУЧЕНИЯ С ПОМОЩЬЮ МЕТОДОВ КРИПТОГРАФИИ И ОБФУСКАЦИИ

***В. А. ПЛАСКОВИЦКИЙ*, П. П. УРБАНОВИЧ**

The article is devoted to the protection of software code from unauthorized use by the methods of cryptography and obfuscation. Provides an overview of the author's program to protect the code and its distinctive features of the analogues

Ключевые слова: защита программного кода, несанкционированное изучение, криптография, обфускация

В современном мире ежегодно взлом программных продуктов приносит ущерб, исчисляемый миллиардами долларов. Наиболее серьезные атаки удаются благодаря изучению программного кода с выявлением скрытых уязвимостей, которые неизбежны нетривиальном приложении [1].

Программный код может находиться в состоянии хранения и выполнения. В первом случае для его защиты лучше подходят криптографические методы, поскольку код можно рассматривать как обычный текст. Во втором случае, когда код должен быть не только защищен от стороннего изучения, но и корректно обрабатываться на ЭВМ, необходимы другие приемы защиты. Одним из них является обфускация, принцип которой основан на изменении программного кода таким образом, чтобы затруднить его анализ потенциальным злоумышленникам, но полностью сохранить выполняемые им функции на ЭВМ [2].

В рамках выполненных исследований была разработана авторская программа «Prok», позволяющая защитить программный код с помощью методов криптографии и обфускации. Отличительными особенностями разработки являются:

1. Модульность защитных алгоритмов, что позволяет разрабатывать их отдельно от самой программы и подключать к ней, регистрируя все необходимые правила взаимодействия с помощью xml-файла. При этом взаимодействие может осуществляться как через текстовые файлы, так и с помощью сокетов, что позволяет размещать их удаленно.

2. Редактор, для составления собственных алгоритмов изменения обрабатываемого текста на основе регулярных выражений. А также дополнительное API, расширяющее возможности составления регулярных выражений для защиты программного кода.

3. Возможность восстанавливать исходный вид кода, даже при использовании необратимых операций обфускации. Достигается за счет ведения специального журнала производимых изменений (ключа), хранящегося в виде xml-файла.

4. Предварительный просмотр производимых изменений, с подстветкой изменяемых участков кода, навигацией по ним и возможностью отмены отдельных из них.

Действие программы можно ограничить отдельным участком кода. Кроме изменений производимых по определенному алгоритму, возможно вставка и удаление данных вручную. Такие действия также отслеживаются в журнале изменений, благодаря чему можно призвести автоматическое восстановление после них. Также в программу встроен генератор ключей, с предоставлением выбора используемых для шифрования символов (или групп символов), основанный на надежном криптографическом генераторе случайных чисел предоставляемом библиотекой NET. Сама программа «Prok» реализована на языке программирования C# (Framework 2.0).

В качестве источника защищаемых данных могут использоваться файлы различных кодировок, определение которой осуществляется нетривиальным способом, в случае неудачи которого можно указать нужную специфическую кодировку вручную.

Литература

1. *Шнаейер, Б*. Секреты и ложь. Безопасность данных в цифровом мире / Б. Шнаейер. – СПб.: Питер, 2003. – 370 с.
2. *Ярмолик В.М*. Криптография, стеганография и охрана авторского права. Монография / В.Н. Ярмолик, С.С. Портянко, С.В. Ярмолик. – Минск, 2007. – 240с.