М. А. Стюгин, *канд. техн. наук, Сибирский государственный университет науки и технологий имени М. Ф. Решетнева, г. Красноярск, styugin@gmail.com*

Технология сигнатурного анализа программного кода с целью защиты от эксплуатации уязвимостей[1]

В данной работе рассмотрена проблема защиты программного кода от эксплуатации уязвимостей, связанных с ошибками разработчиков, отсутствиями проверки корректности входных данных и закладными подпрограммами. Автором предлагается метод, в автоматическом режиме позволяющий обнаруживать аномалии в работе программного кода, которые могут быть ассоциированы с эксплуатацией уязвимостей. Технология позволяет корректировать программный код с целью устранения аномалий и тем самым блокировки возможных атак.

Ключевые слова: информационная безопасность, защита кода, рандомизация кода, сигнатурный анализ, защита от исследования.

Введение

Эксплуатация уязвимостей программного кода на сегодняшний день является одной из наиболее актуальных проблем информационной безопасности. Существует огромное количество сервисов и ресурсов, доступных извне информационных систем. Такие ресурсы могут содержать ошибки разработчиков, специально оставленные закладные подпрограммы или некорректную валидацию входных данных от пользователя.

Для защиты от подобных атак существует три основных направления:

1. Поиск уязвимостей непосредственно в самом программном коде с использованием специальных технологий разработки и инструментов сканирования кода.

2. Анализ пользовательских данных, передаваемых на вход системы на предмет возможных преднамеренных атак.

3. Непрерывное изменение различных параметров информационной системы, без знания которых невозможна или затруднительна реализация атаки.

Третье направление больше известно как Instruction Set Randomization (ISR) и является подклассом систем на основе технологии Moving Target Defense [1–3]. Оно возникло как попытка элиминации рисков, остающихся в результате применения первых двух методов, и исходит из предположения, что атакующему необходимо иметь некую информацию об атакуемой системе, чтобы реализовать уязвимости. Если эту информацию постоянно менять, то и эксплуатация уязвимостей станет затруднительна.

Одним из наиболее распространенных методов ISR является добавление суффиксов в виде случайных n-значных переменных к командам высокоуровневого языка программирования [4–6]. Не зная значения суффикса, злоумышленник не может сгенерировать корректный запрос на инъекцию кода, поскольку команды с неправильным суффиксом не будут выполнены.

[1] Работа выполнена при поддержке гранта РФФИ проект №16-29-09456 офи_м.