УДК 004.052

АНАЛИЗ МЕЖПРОЦЕДУРНЫХ ЗАВИСИМОСТЕЙ НА ОСНОВЕ ГРАФА СВОЙСТВ ПРОГРАММНОГО КОДА

Садырин Д.С.
Университет ИТМО, Санкт-Петербург

Научный руководитель – к.т.н., доцент Дергачев А. М.
Университет ИТМО, Санкт-Петербург

В работе рассматривается подход к межпроцедурному анализу приложений, основанный на графах свойств программного кода. Данный подход применим для поиска ошибок в исходном коде программ на языке C.

Введение. В настоящее время программное обеспечение (ПО) используется в разных областях человеческой деятельности. Ошибки в ПО могут привести к потерям и ущербу, поэтому задача обнаружения ошибок в нем является актуальной. Одним из способов решения этой задачи является статический анализ программного кода.

Основная часть. Анализ программного кода предполагается выполнять на основе граф свойств – направленного мультиграфа с метками на ребрах и парами "ключ-значение" на вершинах или ребрах. Функциями обхода графа могут быть такие функции, как проверка предиката для вершины, получение входящих и исходящих вершин с условиями для меток на ребрах или условиями для свойств. Составные функции можно задавать в виде композиции других функций обхода. Предлагается осуществлять перевод исходного кода для статического анализа, в представление, называемое графом свойств кода, которое объединяет подходы классического анализа программ, а именно абстрактное синтаксическое дерево, граф потока управления и граф зависимостей программ, в общую структуру графа. Работая с полученным представлением, становится возможным идентифицировать такие ошибки в коде программы, как переполнение буфера, целочисленные переполнения, уязвимости строк форматирования или неправильной работы с динамической памятью, путем описания этих ошибок через функции обхода для полученного графа. Также возможно проводить межпроцедурный анализ кода – анализ программы с несколькими процедурами, с учетом того, как информация передается между этими процедурами. Для этого необходимо дополнить полученные графы свойств каждой функции зависимостями по данным между аргументами, переданными в теле вызывающей функции и параметрами вызываемой функции, оператором возврата результата функции и переменной в теле вызывающей функции, также учесть модификацию переменных, переданных по ссылке, внутри вызываемой функции. Граф свойств кода сохраняется в графовую базу данных для последующего задания запросов к свойствам кода. Основные функции обхода графа реализуются с помощью стандартных функций языков запросов к графовым базам данных.

Выводы. Данный подход реализован на фреймворке для статического анализа Joern, который применим для описания и поиска ошибок в исходном коде реализации языка программирования PHP. Преимуществом данного подхода является возможность описывать ошибки в исходном коде программ не только условиями на синтаксические конструкции, но также на поток выполнения и поток данных программы. Это дает возможность находить трудно достижимые ошибки повышая тем самым качество ПО.

Садырин Д.С. (автор) _____
Ильина А. Г. (соавтор) _____
Дергачев А.М. (научный руководитель) _____