**BC/NW 2024№ 1 (41):6.3**

**АВТОМАТИЧЕСКАЯ ГЕНЕРАЦИЯ МОДУЛЬНЫХ ТЕСТОВ ДЛЯ ЯЗЫКА C++ С ИСПОЛЬЗОВАНИЕМ ДИНАМИЧЕСКОГО АНАЛИЗА**

Павлов В.А., Хиль С.Ю., Мирзеабасов О.А.

В современном мире, тестирование программного обеспечения играет важную роль, позволяя оперативно определять некорректное поведение программного кода. Учитывая темпы современной разработки, эффективное тестирование становится особенно актуальным.

Учитывая возросшие требования в области безопасности информации и анализа программного обеспечения, возникла проблема анализа путей выполнения программного кода, при разных входных данных. Lля решения подобных проблем были созданы инструменты символьного исполнения. Они позволяют анализировать различные пути выполнения программы, однако их использование в больших программных комплексах крайне затруднительно. Поскольку подобные анализаторы не имеют возможность отдельно тестировать функции и методы программного комплекса, а могут лишь тестировать весь комплекс целиком.

Решением вышеописанной проблемы, может служить автоматическая генерация модульных тестов с использованием динамического анализа. С помощью генерации модульных тестов, учитывающих результаты работы динамического анализатора, можно значительно повысить эффективность тестирования программ.

В рамках исследования проблемы анализа и тестирования программы с использованием динамического анализа, были использованы следующие инструменты символьного исполнения: Manticore, S2E, Triton и Angr[1]. По результатам анализа готовых программных решений, установлено, что Angr наиболее эффективно справляется с поставленной проблемой, за счет эффективного покрытия путей выполнения программ[1].

Таким образом, задача разработки или модификации готовых решений имеющихся инструментов для динамического анализа программ, ориентированных на обнаружение ошибок и генерацию модульных тестов, является актуальной.

**Литература**

1. **XIE, Bailin and LI, Qi and LUO, Jiabin.** ICIIT '22: Proceedings of the 2022 7th International Conference on Intelligent Information Technology. Association for Computing Machinery “Program Vulnerability Mining System Based on Symbolic Execution” – 02.2022 – С. 83-89.



















