Минобрнауки России
Федеральное государственное учреждение
«Федеральный исследовательский центр
Институт прикладной математики им. М.В. Келдыша
Российской академии наук»
(ИПМ им. М.В. Келдыша РАН)

125047, Москва, Миусская пл., 4 Тел. 8 (499) 220-72-33 Факс 8 (499) 972-07-37
http://keldysh.ru e-mail: office@keldysh.ru
ОКПО 02699381 ОГРН 1037739115787 ИНН/КПП 7710063939/771001001

28.11.2024 № 11103-9422/1132
На № 924-2024 от 17.09.2024

УТВЕРЖДАЮ
Заместитель директора по научной работе
ИПМ им. М.В. Келдыша РАН
член-корр. РАН, д.ф.-м.н, проф.

М.В. Якобовский
28.11.2024

Отзыв
ведущей организации на диссертацию
Саргсяна Севака Сениковича
«Методы оптимизации алгоритмов статического и динамического анализа программ», представленную на соискание ученой степени доктора технических наук по специальности 2.3.5 – математическое и программное обеспечение вычислительных систем, комплексов и компьютерных сетей

Диссертационная работа Саргсяна С.С. посвящена методам статического и динамического анализа программ, а также их универсальной комбинации для поиска сложных для обнаружения дефектов. В работе предложен ряд новых методов для эффективного поиска клонов исходного и бинарного кода, копий известных уязвимостей и ошибок, связанных с использованием динамической памяти. Кроме того, разработаны новые

методы фаззинга для различных сценариев, включая генерацию сложно структурированных данных и интеграцию статического анализа и символьного выполнения с фаззингом. **Актуальность** данной работы обусловлена тем, что, несмотря на рост количества и качества инструментов анализа программного обеспечения, число ежегодно обнаруживаемых ошибок продолжает увеличиваться. Только за последнее десятилетие количество найденных ошибок возросло в три раза.

В диссертации получены следующие **актуальные результаты**, обладающей **научной новизной**:

- архитектура и экспериментальный прототип платформы анализа программ, который обеспечивает сбор артефактов из большого объема открытого ПО и информации об известных уязвимостях, а также предоставляет унифицированный подход к комбинированию различных методов анализа кода в зависимости от поставленных задач;
- масштабируемые и точные методы поиска клонов кода, основанные на обнаружении схожих подграфов максимального размера в графах зависимостей программ, построенных на основе промежуточных представлений исходного и бинарного кода;
- метод сопоставления исходных и бинарных файлов, при котором из исходного кода генерируются бинарные файлы, скомпилированные с различными уровнями оптимизации и содержащие отладочную информацию. Затем производится сопоставление инструкций этих бинарных файлов с помощью разработанного инструмента для поиска клонов бинарного кода, и в завершение выполняется сопоставление исходного кода с инструкциями бинарных файлов на основе отладочной информации;

- метод поиска утечек памяти для языков Си/Си++, который на первом этапе обнаруживает утечки на специальном представлении программы, отражающем поток управления и данных с учетом смещений доступа к указателям и полям структур, а на втором этапе проверяет выполнимость путей ошибок с помощью метода направленного символьного выполнения;
- метод фаззинга программ, который генерирует структурированные данные на основе специализированных автоматов БНФ-грамматик, динамически изменяя их веса в процессе фаззинга. Это позволяет адаптировать шаблоны генерируемых данных в зависимости от их эффективности, что способствует увеличению покрытия кода;
- метод фаззинга интерфейсных функций, позволяющий генерировать цепочки вызовов, где возвращаемые значения одних функций используются в качестве аргументов для других. Это обеспечивает подготовку необходимых ресурсов для тестирования сложных сценариев взаимодействия нескольких функций в среде выполнения;
- метод направленного фаззинга для быстрой генерации входных данных, направленных на выполнение конкретных инструкций или фрагментов целевой программы, содержащих потенциальные уязвимости или дефекты;
- метод интеграции статического анализа с фаззингом, который использует статический анализ для извлечения константных значений, применяемых в условных операторах, а затем использует эти константы для генерации входных данных, обеспечивающих покрытие соответствующих ветвей кода;

Достоверность результатов, полученных в диссертационной работе, обеспечивается множественными экспериментальными исследованиями,

которые были проведены на различных тестовых наборах и проектах. Помимо этого, результаты подтверждены обнаружением и исправлением ошибок в реальных проектах, как с открытым исходным кодом, так и в проприетарных программных продуктах. Широкий диапазон протестированных программных систем свидетельствует о надежности и применимости предложенных методов в условиях реальной разработки.

Теоретическая значимость диссертации заключается в разработанной концепции платформы для анализа программ, а также в методах и алгоритмах статического и динамического анализа, которые продемонстрировали свои преимущества в ходе экспериментальных тестирований по сравнению с существующими решениями.

Практическая значимость работы определяется тем, что на основе предложенных методов была создана программная платформа GENES ISP. Эта платформа включает функционал для сбора артефактов ПО и инструменты, реализующие методы статического и динамического анализа, а также предоставляет возможность комбинированного применения всех инструментов в рамках непрерывной интеграции. GENES ISP уже внедрен в процессе разработки ПО в нескольких организациях. Разработанное средство может быть использовано на всех этапах жизненного цикла создания безопасного ПО, что соответствует требованиям ГОСТ Р 56939-2016 и "Методики выявления уязвимостей и недекларированных возможностей в программном обеспечении" ФСТЭК Российской Федерации. Кроме того, отдельные методы были внедрены в инструменты Svace и ISP-Fuzzer, входящий в состав Crusher, которые считаются индустриальными стандартами в области разработки безопасного ПО.

Анализ содержания работы.

Диссертация включает введение, семь глав, заключение и список литературы из 271 наименования. Общий объем работы составляет 268 страниц, содержащих 56 рисунков и 35 таблиц.

Во введении обосновываются актуальность темы, новизна и практическая значимость диссертационного исследования, а также формируются его цели и задачи. В нем также представлены основные результаты диссертации и приведена ее структура.

В первой главе обосновывается важность исследований в области безопасности программного обеспечения. Анализируются существующие методы и их ограничения, выделяются ключевые направления, включая создание платформы для интеграции инструментов анализа, разработку средств поиска клонов и известных уязвимостей, сопоставление исходного и бинарного кода, выявление ошибок, связанных с форматными строками и динамической памятью, а также оптимизацию методов фаззинга. Также определяются требования к платформе анализа, формулируется ее концепция и описываются основные архитектурные элементы, обеспечивающие эффективную работу с большими объемами ПО и комбинирование методов анализа в зависимости от конкретных задач.

Во второй главе представлен комплексный обзор методов статического и динамического анализа программ, охватывающий поиск клонов кода, сравнение исполняемых файлов, анализ изменений между версиями ПО и методы выявления ошибок.

В третьей главе описаны разработанные методы для поиска клонов кода как в исходных, так и в бинарных файлах, включая инструменты для обнаружения неисправленных ошибок и сопоставления исходного и бинарного кода. Технология основана на графах зависимостей программ, построенных на промежуточном представлении исходного и бинарного кода.

Четвертая глава посвящена методам выявления утечек памяти, анализу проблем некорректного использования динамической памяти и способам обработки помеченных данных. Предлагается метод, который комбинирует статический анализ с направленным символическим выполнением, обеспечивая передовые результаты для поиска утечек памяти.

В пятой главе рассматриваются разработанные методы фаззинга, включая их сочетание с символьным выполнением и статическим анализом. Для статического анализа применяется промежуточное представление исходного кода. Также обсуждаются методы фаззинга для программ, работающих со структурированными данными и интерфейсными функциями. Для генерации сложно структурированных данных используется внутреннее представление платформы ANTLR.

В шестой главе описана интеграционная платформа, созданная для объединения различных методов анализа программ. Рассматриваются функциональные возможности платформы и приводятся примеры ее применения. В частности, комбинация статического анализа и направленного фаззинга позволяет выявлять ошибки в пакетах ОС Debian, что подтверждает эффективность сочетания различных методов и самой платформы.

В седьмой главе подчеркивается практическая значимость работы. Обобщены ошибки, обнаруженные в открытом ПО с использованием новых методов анализа и единой платформы, а также акцентируется внимание на критичности этих ошибок, выявленных в крупных проектах, которые могут затронуть всех пользователей интернета.

В заключении сформулированы основные результаты работы и предложены возможные направления для дальнейших исследований.

Следует отметить ряд **замечаний** к тексту диссертации.

1) Название диссертационной работы представляется излишне общим. Статический и динамический анализ могут использоваться: для выполнения различных компиляторных оптимизаций; в процессе автоматического/автоматизированного распараллеливания программ; для верификации свойств ПО и в других направлениях. В данной работе статический и динамический анализ программ используются для выявления ошибок и уязвимостей в контексте разработки безопасного программного обеспечения, что следовало бы указать в названии работы.

2) В главе 6 ничего не сказано об интеграции разработанной платформы с антивирусным ПО, в котором имеется большое количество вредоносных сигнатур. Использование этих сигнатур было бы очень полезным при анализе больших бинарных кодов на наличие ошибок и вирусов.

3) В главе 4 при анализе текстов программ на C++ на предмет использования освобожденной памяти не учитывается механизм исключений или глубокой рекурсии, при котором факт реального освобождения памяти установить проблематично.

4) В главе 3 методы поиска клонов исходного и бинарного кода описаны весьма поверхностно. Также не определена и не продемонстрирована масштабируемость этих методов.

5) В главе 5 продекларирована возможность применения инструмента ISP-Fuzzer в параллельном режиме, но не приведены результаты исследования ускорения и эффективности распараллеливания. В частности, когда автор приводит время работы инструмента на конкретных задачах фаззинга, ничего не говорится об использованном количестве параллельных процессов.

6) В работе упоминается множество систем, в разработке которых участвовал автор диссертации: инструмент поиска клонов кода для C/C++ программ "CCD"; инструмент поиска клонов кода для бинарных файлов "BINCCD"; инструмент анализа изменений между двумя версиями программы "patchAnalysis"; инструмент фаззинга программ "ISP-Fuzzer", "LibraryIdentifier". Из текста работы не ясно, какие из разработанных инструментов включены в состав платформы, указанной в первом выносимом на защиту результате, или используют предложенную автором архитектуру.

7) В реализованном автором комбинированном методе поиска утечек памяти в алгоритме поиска всех путей между двумя вершинами графа MemoryOperationGraph вводится ограничение PATHS_LIMIT=100000 на

число путей для пары входных и выходных точек. Данное ограничение позволяет использовать алгоритм в случае экспоненциального роста количества путей потока управления. Обоснование выбранного значения PATHS_LIMITS в работе не приводится. Говорится лишь, что "на практике, данное ограничение сработало менее 10 раз".

8) Текст диссертации вынужденно содержит множество сокращений и аббревиатур. Не все из них расшифрованы в тексте (например, "LLVM" и "EC" в главе 3, и т.д.). Целесообразно было включить раздел «Список обозначений и сокращений».

Указанные замечания не влияют на общую высокую оценку диссертационной работы. В целом диссертационная работа Саргсяна С.С. представляет собой самостоятельную и законченную научно-исследовательскую работу, обладающую высокой научной и практической значимостью, решающей важную проблему поиска ошибок и уязвимостей в больших комплексах программ путём статического и динамического анализа. Результаты и выводы, приведенные в диссертации, могут быть использованы при разработке безопасного и доверенного системного и прикладного программного обеспечения в таких организациях, как ИСП им. В.П. Иванникова РАН, ИПМ им. М.В. Келдыша РАН, ИПС им. А.К. Айламазяна РАН, МГУ им. М.В. Ломоносова, НИЦ "Курчатовский институт" – НИИСИ, на предприятиях Росатома и Роскосмоса, в других университетах, научно-исследовательских и промышленных профильных организациях.

Основные результаты диссертации опубликованы в ведущих российских и зарубежных рецензируемых изданиях и прошли апробацию на международных и всероссийских конференциях. По теме диссертации автором опубликовано 24 работы. Автореферат правильно отражает содержание диссертации и ее основные результаты.

требованиям «Положения о присуждении ученых степеней», а Саргсян Севак Сеникович, заслуживает присуждения ему учёной степени доктора технических наук по специальности 2.3.5 – математическое и программное обеспечение вычислительных систем, комплексов и компьютерных сетей.

Отзыв обсужден на расширенном заседании отдела «Программное обеспечение высокопроизводительных вычислительных систем и сетей» ИПМ им. М.В. Келдыша РАН, протокол № 1 от 22 ноября 2025 года.

Отзыв составил:

Ведущий научный сотрудник ИПМ РАН,

доктор физ.-мат. наук _____/Сергей Владимирович Поляков/

28 ноября 2024 года