Анализ кода внешних компонентов проекта для поиска фрагментов обработки секретов с целью уточнения находок сканеров поиска секретов

Ермакова Анна Ивановна

Студент

Факультет ВМК МГУ имени М. В. Ломоносова, Москва, Россия

E-mail: anyaerm2003@yandex.ru

Научный руководитель *— Самосадный Кирилл Алексеевич*

Наличие чувствительных данных в исходном коде приложения может привести к их компрометации при раскрытии исходного кода, например, в системах контроля версий [1]. К категории чувствительных данных, называемых секретами, в рамках данной работы относятся токены доступа, ключи шифрования и пароли, раскрытие которых несет угрозы защищенности. Существующие инструменты поиска оставленных секретов в исходном коде находят значительное количество потенциальных секретов, часть из которых может быть ложными срабатываниями [2-3]. Поэтому актуальна проблема приоритизации находок инструментов поиска секретов на основе дополнительного анализа исходного кода.

В данной работе предлагается подход к приоритизации, основанный на проверке использования найденного секрета в исходном коде. Для этого используется анализ помеченных данных (taint analysis), где источником является объявление секрета, а приемником — потенциальная конструкция его обработки. В ходе анализа возможных сценариев работы с секретами была выделена классификация этапов обработки секретов. Она включает ввод секретов в систему, их преобразование и последующее использование при взаимодействии с внешними сервисами. Информация об источниках берется из результатов работы инструментов поиска секретов, а набор приемников требуется сформировать на основе этапов обработки секретов в коде.

В рамках метода уточнения предлагается формирование базы приемников вручную путем отбора популярных библиотек для работы с секретами, охватывающих все этапы обработки в соответствии с классификацией. Такой подход требует значительных затрат времени на ручной анализ кода. Для решения данного недостатка в работе предлагается модификация метода с использованием алгоритма

выявления потенциальных конструкций обработки секретов с помощью статического анализа исходного кода библиотек. Методы статического анализа используются для поиска в коде идентификаторов конструкций-приемников, поиск которых основан на использовании признаков: типе конструкции, именах и контексте конструкций — что позволяет сопоставить найденные фрагменты с конкретным этапом обработки секрета.

Итоговый процесс приоритизации секретов состоит из двух этапов. На первом этапе автоматизированно строится база приемников под конкретный проект на основе используемых в проекте библиотеках. Второй этап реализует уточнение секретов с учетом набора приемников, сформированного на предыдущем этапе. Тестирование предложенного двухэтапного метода проведено путем сканирования проекта, разработанного на языке Java.

Литература

1. Feng R. et al. Automated detection of password leakage from public GitHub repositories // Proceedings of the 44th International Conference on Software Engineering. 2022. – C.175–186.
2. Meli M., McNiece M. R., Reaves B. How bad can it git? characterizing secret leakage in public github repositories //NDSS. – 2019.
3. Saha A. et al. Secrets in source code: Reducing false positives using machine learning //2020 International Conference on COMmunication Systems & NETworkS (COMSNETS). – IEEE, 2020. - C.168–175.