Улучшение инструментов для отладки приложений Android и исследование возможностей их применения для анализа поведения ПО

Сидоров Владислав Олегович[1,2]

Научный руководитель: Проскурин Вадим Геннадьевич[2]

[1]МФТИ [2]ИСП РАН

20 мая 2025 г.

План

1. Введение
2. Постановка задачи
3. Проделанная работа
4. Результаты

Введение

Мотивация

Хотим понимать поведение программы во время ее исполнения

Введение

Мотивация

Хотим понимать поведение программы во время ее исполнения

Зачем это может быть нужно:

Введение

Мотивация

Хотим понимать поведение программы во время ее исполнения

Зачем это может быть нужно:

- Отладка

Введение

Мотивация

Хотим понимать поведение программы во время ее исполнения

Зачем это может быть нужно:

- Отладка
- Обнаружение подозрительной активности

Введение

Мотивация

Хотим понимать поведение программы во время ее исполнения

Зачем это может быть нужно:

- Отладка
- Обнаружение подозрительной активности
- Проверка соблюдения требований к ПО

Решение

Одним из подходов для решения этой задачи является отслеживание вызовов некоторого набора методов во время исполнения программы (call monitoring)

Аналогия

Широко известный **ptrace**

- Классический системный вызов на Unix-подобных системах
- Позволяет одному процессу на низком уровне контролировать работу другого процесса
- Чаще всего используется для внедрения своего кода в чужой процесс (в т.ч. для call monitoring)

Android

Почему платформой выбран Android?

Android

Почему платформой выбран Android?

- Огромное количество приложений. Потенциальный доступ к чувствительной информации пользователя

Android

Почему платформой выбран Android?

- Огромное количество приложений. Потенциальный доступ к чувствительной информации пользователя
- Основная масса ПО на Java $\Rightarrow$ имеем одинаковую основную структуру всех приложений (пакеты)

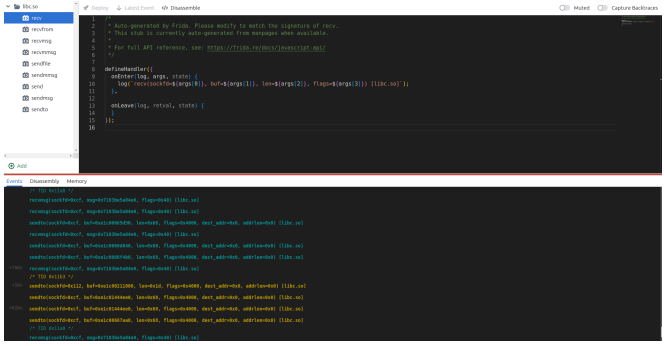
Задача

Разработать инструмент общего назначения для автоматизированного динамического анализа поведения ПО на Android методом трассировки вызовов

Сравнение существующих решений

Готовые инструменты есть, но у них всех присутствуют критически важные для использования недостатки

frida-trace

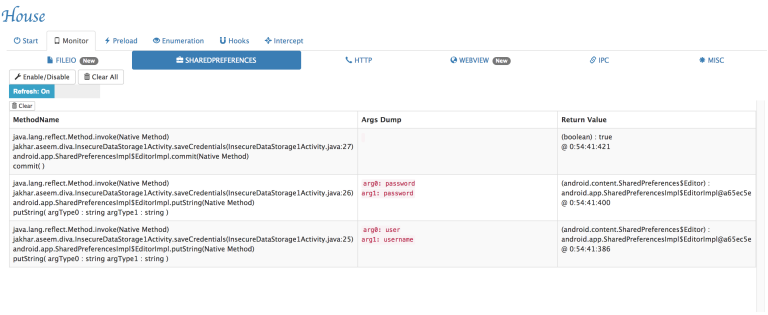


`https://github.com/frida/frida`

Недостатки:

- Результаты анализа не сохраняются, отсутствует возможность загрузить отчет
- Во время работы инструмента нельзя никак менять список отслеживаемых методов
- При каждом запуске нужно указывать каждый метод/пакет отдельно

House

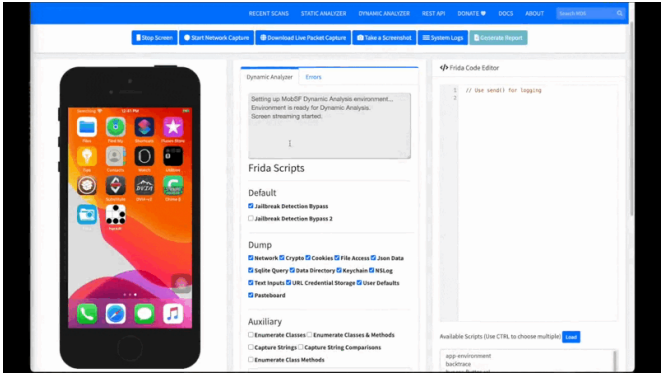


`https://github.com/nccgroup/house`

Недостатки:

- Отсутствие поддержки (последнее изменение — 5 лет назад)
- Результаты анализа не сохраняются, отсутствует возможность загрузить отчет
- Перехватываемые методы Android API вшиты в код инструмента + их мало

MobSF



`https://github.com/MobSF/Mobile-Security-Framework-MobSF`

Недостатки:

- Динамический анализ только после полного статического анализа
- Отсутствие возможности использовать с версией Android > 11
- Перехватываемые методы Android API вшиты в код инструмента

- Open-source фреймворк для инструментации приложений (не только под Android)
- Огромные возможности по внедрению своего кода в чужие приложения
- Работает через скрипты на JavaScript, есть свой API для Java

Техническое задание

- Работаем только с Java-методами
- За основу берем фреймворк для динамической инструментации Frida
- Устраняем недостатки существующих решений

Проделанная работа

Список методов

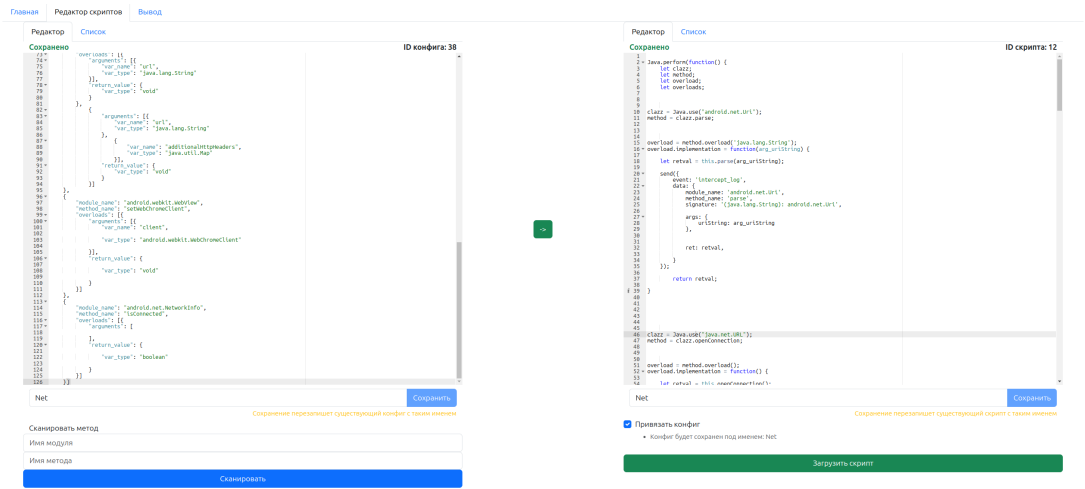
Был подготовлен список интересующих методов для трассировки

org.apache.http.impl.client.HttpClient.execute(HttpUriRequest)
io.socket.client.Socket.connect()
java.net.InetAddress.getHostAddress()
android.media.ImageReader.getSurface()
android.hardware.Camera.open(int)
android.hardware.camera2.CameraCaptureSession.capture(CaptureRequest, CameraCaptureSession.CaptureCallback, Handler)
android.net.wifi.WifiManager.getConnectionInfo()
java.security.X509Certificate.getPublicKey()
com.google.firebase.database.DatabaseReference.child(String)
android.webkit.WebView.setWebChromeClient(WebChromeClient)
android.media.MediaRecorder.start()
android.media.projection.MediaProjectionManager.createScreenCaptureIntent()
android.media.AudioManager.setStreamVolume(int, int, int)
android.location.Location.getLatitude()
android.location.LocationManager.requestLocationUpdates(String, long, float, LocationListener)
java.security.MessageDigest.getInstance(String)
...

Всего порядка 80 функций

Прототип

Был разработан полноценный прототип инструмента



Что умеет прототип?

- Подключение к Android устройству и определенному процессу на нем

Что умеет прототип?

- Подключение к Android устройству и определенному процессу на нем
- Сохранение скриптов перехвата для будущего использования

Что умеет прототип?

- Подключение к Android устройству и определенному процессу на нем
- Сохранение скриптов перехвата для будущего использования
- Динамическая загрузка любого количества скриптов

Что умеет прототип?

- Подключение к Android устройству и определенному процессу на нем
- Сохранение скриптов перехвата для будущего использования
- Динамическая загрузка любого количества скриптов
- Сохранение логов перехвата в ожидаемом формате в БД, формирование отчета

Что умеет прототип?

- Подключение к Android устройству и определенному процессу на нем
- Сохранение скриптов перехвата для будущего использования
- Динамическая загрузка любого количества скриптов
- Сохранение логов перехвата в ожидаемом формате в БД, формирование отчета

Учли недостатки существующих инструментов

Тестирование

Инструмент был протестирован на реальном вредоносном приложении - **AndroidRAT**
Данная программа позволяет получить удаленный доступ к устройству через консоль. В консоли можно выполнять заранее заготовленные команды (получить фото с камеры, записать аудио)
Разработанный инструмент успешно перехватывает методы Android API, соответствующие командам

Результаты

Разработан прототип инструмента для динамического анализа поведения ПО
Тестирование показало, что инструмент корректно работает и пригоден для практического применения
В прототипе были устранены недостатки существующих решений

Дальнейшие перспективы

- Расширение функционала по сбору данных
- Внедрение автоматизированного анализа данных
- Защита от обнаружения инструментации

Конец