Аннотация – в документе представлен анализ статьи " MalPurifier: Enhancing Android Malware Detection with Adversarial Purification against Evasion Attacks". Анализ посвящён различным аспектам статьи, включая используемую методологию, экспериментальную установку и полученные результаты.

Этот анализ представляет собой качественное изложение документа, предлагающее ценную информацию специалистам в области безопасности, исследователям и практикам в различных областях. Понимая сильные стороны и ограничения платформы MalPurifier, заинтересованные стороны смогут лучше оценить её потенциальные применения и вклад в совершенствование систем обнаружения вредоносных программ Android. Анализ особенно полезен для тех, кто занимается кибербезопасностью, машинным обучением и безопасностью мобильных приложений, поскольку в нем освещаются инновационные подходы к снижению рисков, связанных с атаками с целью предотвращения обнаружения.

I. Введение

В документе под названием "MalPurifier: Enhancing Android Malware Detection with Adversarial Purification against Evasion Attacks" представлен новый подход к улучшению обнаружения вредоносных программ для Android, особенно в условиях состязательных атак уклонения (adversarial evasion attacks). В документе подчёркивается, что это первая попытка использовать состязательную очистку для смягчения атак в экосистеме Android, предоставляя многообещающее решение для повышения безопасности систем обнаружения вредоносных программ Android.

A. Мотивация:

- **Распространённость вредоносного ПО для Android**: В документе освещается широко распространённая проблема вредоносного ПО для Android, которое представляет значительные угрозы безопасности для пользователей и устройств.
- **Методы уклонения**: часто используются методы уклонения для модификации вредоносных программ, что затрудняет их идентификацию традиционными системами обнаружения.

B. Проблемы:

- **Состязательные атаки**: обсуждаются проблемы, связанные с состязательными атаками, когда небольшие изменения кода вредоносных программ позволяют избежать обнаружения.
- **Уязвимости системы обнаружения**: Существующие системы обнаружения вредоносных программ уязвимы для этих состязательных атак, что требует более надёжных решений.

C. Цель и предлагаемое решение:

- **Повышение надёжности обнаружения**: цель исследования – повышение устойчивость систем обнаружения вредоносных программ Android к атакам с использованием состязательного уклонения.
- **Предлагаемое решение**: MalPurifier, направлено на очистку мусора в образцах и восстановление вредоносного ПО до обнаруживаемой формы.
- **Используемые методы**: В системе используются такие методы, как автокодирование и генеративные состязательные сети (GAN) для процесса очистки.

D. Техники, используемые при атаках уклонения:

- **Образцы состязательности**: часто используются методы уклонения для модификации вредоносных программ, что затрудняет их идентификацию традиционными системами обнаружения.
- **Обфусцирование**: Такие методы, как шифрование кода, упаковка и полиморфизм, используются для изменения внешнего вида вредоносного ПО без изменения его функциональности.

E. Значение:

- **Улучшенная безопасность**: Расширяя возможности систем обнаружения вредоносных программ, MalPurifier стремится обеспечить лучшую безопасность устройств Android.
- **Вклад в исследование**: Статья вносит свой вклад, устраняя пробел в надёжных решениях для обнаружения вредоносных программ, способных противостоять злоумышленным атакам.

F. Преимущества

- **Высокая точность**: MalPurifier демонстрирует высокую эффективность, достигая точности более 90,91% при 37 различных атаках. Это указывает на высокую производительность при обнаружении вредоносных программ.

- **Масштабируемость**: Метод легко масштабируется для различных моделей обнаружения, обеспечивая гибкость и надёжность в его реализации, не требуя значительных модификаций.
- **Лёгкий и гибкий**: Использование модели с шумоподавляющим автоэнкодером (Denoising AutoEncoder, DAE) обеспечивает лёгкий и гибкий подход к очистке от вредоносного ПО. Это гарантирует, что метод может быть интегрирован в существующие системы с минимальными накладными расходами.
- **Комплексная защита**: Уделяя особое внимание очистке от вредоносных программ, MalPurifier устраняет критическую уязвимость в системах обнаружения вредоносных программ на основе ML, повышая их общую безопасность и устойчивость к изощренным методам уклонения.

G. Ограничения

- **Обобщение на другие платформы**: Текущая реализация и оценка сосредоточены исключительно на экосистеме Android. Эффективность MalPurifier на других платформах, таких как iOS или Windows, остаётся непроверенной и неопределённой.
- **Проблемы с масштабируемостью**: хотя в документе утверждается о масштабируемости, фактическая производительность и действенность MalPurifier в крупномасштабных сценариях обнаружения в реальном времени тщательно не оценивались. Это вызывает вопросы о практической применимости в средах с соответствующими сценариями нагрузки.
- **Вычислительные издержки**: Процесс очистки приводит к дополнительным вычислительным издержкам. Несмотря на то, что он описывается как лёгкий, его влияние на производительность системы, особенно в средах с ограниченными ресурсами требует дальнейшего изучения.
- **Адаптация к состязательности**: могут разрабатываться новые стратегии для адаптации к процессу очистки, потенциально обходя средства защиты, предоставляемые MalPurifier. Постоянная адаптация и совершенствование методов необходимы для своевременного опережения угроз.
- **Показатели оценки**: Оценка в первую очередь фокусируется на точности обнаружения и устойчивости к атакам уклонения. Другие важные показатели, такие как потребление энергии, опыт работы с пользователем и долгосрочная эффективность, не учитываются, что ограничивает полноту оценки.
- **Интеграция с существующими системами**: В документе подробно не обсуждается интеграция MalPurifier с существующими системами обнаружения вредоносных программ и потенциальное влияние на их производительность. Необходимы бесшовные стратегии интеграции и комбинированные оценки эффективности

H. Влияние на технологию

- **Прогресс в обнаружении вредоносных программ**: MalPurifier представляет собой значительный технологический прогресс в области обнаружения вредоносных программ. Используя методы состязательной очистки, он повышает устойчивость систем обнаружения вредоносных программ Android к атакам-уклонениям. Это нововведение может привести к разработке более безопасных и надёжных инструментов обнаружения вредоносных программ.
- **Механизмы защиты от состязательности**: Статья вносит вклад в более широкую область состязательного машинного обучения, демонстрируя эффективность состязательной очистки. Метод может быть адаптирован к другим областям кибербезопасности, таким как обнаружение сетевых вторжений и защита конечных точек, повышая общую устойчивость систем к новым атакам.
- **Приложения для машинного обучения**: Использование шумоподавляющих автоэнкодеров (DAE) и генеративных состязательных сетей (GAN) в MalPurifier демонстрирует потенциал передовых моделей машинного обучения в приложениях кибербезопасности. Это может вдохновить на дальнейшие исследования и разработки по применению этих моделей к другим задачам безопасности, таким как обнаружение фишинга и предотвращение мошенничества.

I. Влияние на отрасль

- **Повышенная безопасность мобильных устройств**: Отрасли, которые в значительной степени зависят от мобильных устройств, такие как здравоохранение, финансы и розничная торговля, могут извлечь выгоду от применения MalPurifier, как следствие, могут лучше защищать конфиденциальные данные и поддерживать целостность мобильных приложений.
- **Снижение числа инцидентов, связанных с кибербезопасностью**: Внедрение надёжных систем обнаружения вредоносных программ, таких как MalPurifier, может привести к сокращению инцидентов кибербезопасности, таких как утечка данных и атаки программ-вымогателей, а также значительной экономии средств для бизнеса и снижению вероятности репутационного ущерба.
- **Преимущества соблюдения нормативных требований**: Расширенные возможности обнаружения вредоносных программ могут помочь организациям соблюдать нормативные требования, связанные с защитой данных и кибербезопасностью. Например, отрасли, подпадающие под действие таких нормативных актов, как GDPR или HIPAA, могут использовать MalPurifier для обеспечения соответствия строгим стандартам безопасности.

- **Инновации в продуктах кибербезопасности**: Компании, занимающиеся кибербезопасностью, могут внедрять методы, представленные в документе, в свои продукты, что приведёт к разработке решений безопасности следующего поколения для повышения конкурентного преимущества на рынке и стимулировать инновации в индустрии кибербезопасности.
- **Межотраслевые приложения**: хотя в статье основное внимание уделяется обнаружению вредоносных Android-программ, основополагающие принципы состязательной очистки могут применяться в различных отраслях. Такие секторы, как производство, государственное управление и транспорт, которые также подвержены воздействию вредоносных программ, могут адаптировать эти методы для усиления своих мер кибербезопасности.

II. Последствия вредосного кода Android для отраслей

- **Производство**: Производственный сектор сильно страдает от кибератак и инцидентов с вредоносными программами. Согласно отчёту Security Navigator 2023 высокий процент инцидентов происходит внутри компании.
- **Государственное управление**: Государственное управление сталкивается с многочисленными инцидентами, приписываемыми внутренним источникам, будь то преднамеренными или случайными.
- **Малые и средние предприятия (МСП)**: МСП особенно уязвимы для атак, при этом высокий процент 49% инцидентов связан с вредоносным ПО согласно Security Navigator за 2023 год.
- **Здравоохранение и социальная помощь**: Сектор здравоохранения также подвержен вредоносным программам, на исправление ИТ-уязвимостей в среднем уходит 491 день.
- **Транспорт и логистика**: сектор сталкивается со значительными проблемами, связанными с вредоносными программами, на исправление которых требуется в среднем 473 дня.
- **Мобильные устройства и Интернет вещей**: Устройства Android, включая смартфоны, умные часы, телевизоры и другие устройства Интернета вещей, часто становятся мишенями вредоносных программ. Компания Trend Micro обнаружила вредоносное ПО, предварительно установленное на заводских устройствах, от которого пострадали по меньшей мере 10 OEM-производителей и потенциально более 40 поставщиков. Google Play также был источником вредоносного ПО: различные вредоносные приложения загружались миллионы раз в разных регионах и отраслях.
- **Слежка и шпионское ПО**: Индустрия, включая такие фирмы, как Cy4Gate / ELT Group, RCS Labs и другие, нацелена на шпионские программы для устройств Android. Эти фирмы занимаются различными видами деятельности, затрагивая широкий спектр платформ и отраслей

III. Исследование

A. *Обнаружение вредоносных программ на основе ML*

Приведённые аспекты обеспечивают понимание роли и задач машинного обучения в обнаружении вредоносных программ, определяют контекст для предлагаемой системы MalPurifier и того, как методы машинного обучения получили широкое распространение для обнаружения вредоносных программ благодаря их способности извлекать закономерности из данных и обобщать их на новые, неизвестные образцы

1) Типы функций

- **Извлечение функций**: подчёркивается важность извлечения функций при обнаружении вредоносных программ на основе ML, где функции являются производными от различных аспектов приложений Android, таких как разрешения, вызовы API и сетевой трафик.
- **Статические функции**: они извлекаются из кода без его выполнения. Образцы включают разрешения, вызовы API и структуру кода.
- **Динамические функции**: они получаются путём анализа поведения приложения во время выполнения, например системные вызовы, сетевую активность и использование памяти.

2) Распространённые алгоритмы ML

- **Контролируемое обучение**: в нем подчёркивается использование алгоритмов контролируемого обучения, таких как деревья принятия решений, SVM и нейронные сети, которым для обучения требуются разметка наборов данных.
- **Неконтролируемое обучение**: упоминаются методы неконтролируемого обучения, такие как кластеризация, которые не требуют разметки данных и могут использоваться для идентификации новых вредоносных программ.

3) Преимущества детекции на основе ML

- **Высокая точность**: ML-методы позволяют достичь высокой точности обнаружения за счёт изучения сложных закономерностей в данных.
- **Адаптивность**: адаптация к новым вредоносным программам путём «перенастройки моделей» с использованием обновлённых наборов данных.

B. *Уклонение от атак*

Приведённые аспекты обеспечивает понимание того, как концептуализируются и выполняются атаки-уклонения, подчёркивая проблемы, с которыми сталкиваются системы обнаружения вредоносных программ при защите от них.

1) *Уклонение от Атак*

- **Определение**: Атаки-уклонения — это стратегии, используемые для модификации вредоносного ПО таким образом, чтобы оно могло обойти обнаружение системами обнаружения вредоносного ПО на основе машинного обучения.
- **Воздействие**: Эти атаки представляют угрозу эффективности систем обнаружения, поскольку они могут привести к необнаруженному заражению вредоносными программами.

2) *Принцип тактики*

- **Образцы состязательности**: Основным принципом, лежащим в основе атак уклонения, является создание примеров состязательности. Это входные данные, специально созданные для того, чтобы обмануть модель машинного обучения и заставить её делать неверные прогнозы.
- **Нарушения**: Состязательные образцы создаются путём добавления небольших, часто незаметных изменений к исходным образцам вредоносных программ для использования уязвимостей в границах принятия решений моделью.
- **Задача оптимизации**: Создание состязательных примеров оформлено как задача оптимизации, цель которой – найти минимальное изменение, которое заставляет модель неправильно классифицировать входные данные.

3) *Методы атаки*

- **Атаки с использованием "белого ящика"**
  - **Определение**: при атаках методом белого ящика злоумышленник обладает полной информацией о целевой модели, включая её архитектуру, параметры и обучающие данные.
  - **Методы**: Распространённые методы включают метод FGSM и PGD, которые используют информацию для создания состязательных примеров.
- **Атаки с использованием черного ящика:**
  - **Определение**: при атаках с использованием черного ящика злоумышленник ничего не знает о целевой модели. Вместо этого они могут только запрашивать модель и наблюдать за её результатами.
  - **Методы**: Методы включают атаки на основе запросов, при которых злоумышленник итеративно запрашивает модель для сбора информации и создания состязательных примеров, при которых состязательные образцы, созданные для одной модели, используются против другой модели.
- **Атаки с использованием "серого ящика":**
  - **Определение**: Атаки методом серого ящика предполагают частичное знание целевой модели, например, её архитектуры, но не её параметров.
  - **Методы**: Эти атаки часто сочетают элементы стратегий как "белого ящика", так и "черного ящика" для создания примеров состязательности.
- **Образцы методов уклонения:**
  - **Манипуляция с признаками**: изменение признаков, используемых моделью обнаружения, например добавление полезных признаков или обфускация вредоносных.
  - **Обфускация кода**: шифрование кода, упаковка и полиморфизм, позволяющие изменить внешний вид вредоносного ПО без изменения его функциональности.

C. Состязательное Очищение

Эти ключевые моменты дают обзор роли и реализации состязательной очистки в защите от атак-уклонений в контексте обнаружения вредоносных программ Android.

1) *Концепция состязательного очищения*

- **Определение**: Состязательная очистка относится к процессу преобразования состязательных в исходную форму перед отправкой в систему обнаружения вредоносных программ.
- **Цель**: Основная цель состоит в устранении враждебных изменений, которые были добавлены для предотвращения обнаружения, восстанавливая выборку до состояния, в котором она может быть точно классифицирована моделью обнаружения.

2) *Методы состязательного очищения*

- **Автоэнкодеры**: нейронные сети, предназначенные для изучения сжатого представления входных данных и их последующей реконструкции. Их можно обучить устранять модификации, путём сопоставления модифицированных версий с их чистыми аналогами.
- **Порождающие состязательные сети (GAN)**: GAN состоят из генератора и дискриминатора. Генератор учится создавать очищенные версии примеров, в то время как дискриминатор различает реальные (чистые) и поддельные (состязательные) образцы. Благодаря этому генератор улучшает свою способность очищать состязательные образцы.
- **Методы шумоподавления**: различные методы обработки сигналов, которые могут применяться для удаления шума из входных данных, тем самым смягчая воздействие враждебных модификаций.

3) *Преимущества состязательной очистки*

- **Независимость от модели**: Состязательная очистка применяется в качестве этапа предварительной обработки, что делает её независимой от конкретной используемой модели обнаружения вредоносного ПО. Это позволяет интегрировать его с различными

системами обнаружения, не требуя внесения изменений в сами модели.

- **Повышенная надёжность**: эффективно устраняя враждебные помехи, состязательная очистка повышает надёжность систем обнаружения вредоносных программ, делая их более устойчивыми к атакам-уклонениям.

4) *Проблемы*

- **Эффективность**: Эффективность состязательной очистки зависит от способности метода очистки точно удалять модификации кода без изменения исходных характеристик образца вредоносного ПО.
- **Вычислительные накладные расходы**: Реализация состязательной очистки может привести к дополнительным вычислительным затратам, которые необходимо сбалансировать с точки зрения точности и надёжности обнаружения.

5) *Направленность исследования*

- **Оптимизация**: Текущие исследования направлены на оптимизацию состязательных методов очистки для достижения баланса между эффективностью очистки и вычислительной эффективностью.
- **Интеграция**: интеграция антивирусной защиты с существующими конвейерами обнаружения вредоносных программ для повышения общей безопасности системы.

D. Модель угроз

Основная цель злоумышленника – создать образцы, которые могут «ускользнуть» от обнаружения системой обнаружения вредоносных программ Android".

1) *Знание противника*

- **Сценарий "белого ящика"**: злоумышленник обладает полной информацией о модели обнаружения вредоносного ПО, включая её архитектуру, параметры и обучающие данные.
- **Сценарий "черного ящика"**: злоумышленник не имеет прямого представления о модели, но может запрашивать её и наблюдать за результатами для получения информации.
- **Сценарий "серого ящика"**: злоумышленник частично осведомлён о модели, например, о её архитектуре, но не о её параметрах.

2) *Возможности противника*

- Злоумышленник формирует образцы, добавляя искажения к исходным образцам вредоносного ПО чтобы избежать обнаружения.
- Злоумышленник может использовать различные методы для создания этих состязательных примеров, включая методы исходя из сценариев упомянутых ранее.

3) *Типы атак*

- **Атаки-уклонения**: Основное внимание уделяется атакам-уклонениям, при которых злоумышленник стремится модифицировать образцы вредоносных программ, чтобы избежать обнаружения без изменения их вредоносных функций.
- **Ограничения**: Злоумышленник ограничен необходимостью минимизировать модификации, чтобы сохранить функциональность и внешний вид исходного вредоносного ПО.

4) *Предположения*

- Предполагается, что система обнаружения представляет модель на основе машинного обучения.
- Предполагается, что противник обладает способностью формировать образцы, используя описанные знания и методы.

5) *Защитный механизм*

Модель угроз создаёт основу для оценки эффективности MalPurifier, целью которого является устранение примеров враждебности и приведение их к форме, которая может быть точно обнаружена системой обнаружения вредоносных программ.

E. Формулировка и архитектура защиты

Основная цель формулировки защиты – разработать систему, которая может эффективно очищать образцы, тем самым повышая надёжность систем обнаружения вредоносных программ Android.

1) *Основа для MalPurifier*

- **Архитектура**: Платформа MalPurifier состоит из двух основных компонентов: модуля очистки и модуля обнаружения.
- **Модуль очистки**: Этот модуль отвечает за удаление нежелательных модификаций из входных выборок. Для достижения этой цели используются такие методы, как автокодирование и генеративные состязательные сети (GAN).
- **Модуль обнаружения**: после очистки очищенные образцы передаются в модуль обнаружения, который представляет собой систему обнаружения вредоносных программ на основе машинного обучения.

2) *Методы очистки*

- **Автоэнкодеры**: Они используются для изучения сжатого представления входных данных и их последующей реконструкции, эффективно удаляя помехи, создаваемые конкуренцией.
- **Генеративные состязательные сети (GAN)**: GAN используются для генерации очищенных версий состязательных примеров. Генератор учится выдавать чистые выборки, в то время как дискриминатор различает реальные (чистые) и состязательные (с модификацией) выборки.

3) *Тренировочный процесс*

- **Обучение**: Модуль очистки обучается на примерах состязательности, чтобы научиться эффективно устранять помехи.
- **Оптимизация**: Процесс обучения включает оптимизацию функций, которые измеряют разницу между очищенными и исходными чистыми образцами, гарантируя, что процесс очистки не изменит невредоносных характеристик образцов.

4) *Рабочий процесс*

- **Обработка входных данных**: Входные выборки, которые могут включать состязательные образцы, сначала обрабатываются модулем очистки.
- **Очистка**: Модуль очистки удаляет нежелательные модификации из входных выборок.
- **Обнаружение**: очищенные образцы затем подаются в модуль обнаружения, который классифицирует

5) *Показатели оценки*

Эффективность системы MalPurifier оценивается с использованием таких показателей, как точность обнаружения, частота ложных срабатываний и устойчивость к атакам противника.

6) *Интеграция с системами обнаружения*

Очищенные образцы поступают в существующие системы обнаружения вредоносных программ, которые затем могут точно классифицировать их, не будучи обманутыми враждебными вмешательствами.

7) *Преимущества*

- **Не зависит от модели**: Процесс очистки не зависит от конкретной модели обнаружения вредоносного ПО, что позволяет интегрировать его с различными системами обнаружения.
- **Повышенная надёжность**: Устраняя враждебные помехи, MalPurifier повышает надёжность систем обнаружения вредоносных программ, делая их более устойчивыми к атакам-уклонениям.

F. Разнообразные состязательное модификации

Идея заключается в формировании различных модификаций, чтобы гарантировать, что модуль очистки сможет обрабатывать широкий спектр стратегий атаки.

1) *Формирование модификаций*

- **Множественные методы атаки**: подчёркивается использование нескольких методов состязательной атаки для создания разнообразного набора примеров состязательности, включая методы атаки как с использованием белого и черного ящика
- **Комбинация методов**: Комбинируя различные методы атаки, система может генерировать полный набор примеров состязательности, которые охватывают различные тактики уклонения, используемые злоумышленниками.

2) *Обучение с различными модификациями*

- **Надёжность обучения**: Модуль очистки обучается с использованием разнообразного набора состязательных примеров, что гарантирует, что модуль научится эффективно устранять широкий спектр модификаций.
- **Улучшенное обобщение**: Обучение с использованием разнообразных модификаций помогает модулю очистки обобщать новые, неизвестные образцы, повышая общую надёжность.

3) *Оценка*

- **Эффективность**: Эффективность использования разнообразных модификаций оценивается путём тестирования модуля очистки на различных типах примеров. Результаты показывают эффективность при обучении на различных модификациях.
- **Точность**: использование разнообразных модификаций приводит к повышению точности обнаружения и надёжности системы обнаружения вредоносных программ.

4) *Преимущества*

- **Комплексная защита**: при включении широкого спектра враждебных воздействий модуль очистки может защищаться от множества стратегий атак, что делает его комплексным механизмом защиты.
- **Повышенная надёжность**: диверсифицированный подход значительно повышает надёжность обнаружения вредоносных программn

G. Добавление защитного шума

Идея состоит в том, чтобы ввести определённый тип шума во входные выборки, чтобы нейтрализовать воздействие враждебных модификаций. Этот защитный шум предназначен для нейтрализации враждебного шума, создаваемого злоумышленниками.

1) *Механизм*

- **Процесс введения шума**: Защитный шум добавляется к входным образцам перед их обработкой модулем очистки. Этот шум тщательно обработан, чтобы устранить враждебные модификации без существенного изменения положительных характеристик образцов.
- **Нейтрализация противоборства**: Вводимый шум направлен на нейтрализацию противоборствующих модификаций, облегчая модулю очистки устранение любых оставшихся противоборствующих воздействий.

2) *Тренировка с защитным шумом*

- **Надёжное обучение:** Модуль очистки обучается на выборках, которые включают как враждебные модификации, так и защитный шум.
- **Улучшенное обучение**: благодаря включению защитного шума во время обучения модуль очистки может лучше научиться очищать состязательные образцы, что приводит к повышению надёжности.

3) Оценка

- **Эффективность**: Эффективность введения защитного шума оценивается путём тестирования модуля очистки на примерах с защитным шумом и без него. Результаты показывают, что модуль работает лучше, когда используется защитный шум.
- **Точность обнаружения**: использование защитного шумоподавления приводит к повышению точности обнаружения и надёжности системы обнаружения вредоносных программ.

4) Преимущества

- **Повышенная надёжность**: Внедрение защитного шума значительно повышает надёжность системы обнаружения вредоносных программ, делая её более устойчивой к атакам противника.
- **Дополнительная защита**: метод дополняет другие защитные механизмы: состязательное очищение и разнообразные модификации, обеспечивая дополнительный уровень безопасности.

5) Проблемы

- **Калибровка по шуму**: Одна из проблем заключается в правильной калибровке защитного шума, чтобы он эффективно нейтрализовал враждебные модификации без снижения производительности системы обнаружения невредоносных образцов.
- **Вычислительные издержки**: Введение защитного шума и управление им могут привести к дополнительным вычислительным затратам, которые необходимо сбалансировать с преимуществами повышенной надёжности

H. Точное извлечение образца

Основная цель – восстановить модифицированные образцы в их исходное состояние, гарантируя, что система обнаружения вредоносных программ сможет их точно классифицировать.

1) Методы извлечения образцов

- **Автоэнкодеры**: Автоэнкодеры используются для изучения сжатого представления входных данных и их последующей реконструкции, эффективно удаляя помехи и восстанавливая исходную выборку.
- **Генеративные состязательные сети (GAN)**: GAN используются для генерации очищенных версий состязательных примеров. Генератор учится выдавать чистые выборки, в то время как дискриминатор различает реальные (чистые) и состязательные (с модификацией) выборки.

2) Тренировочный процесс

- **Обучение**: Модели восстановления обучаются на примерах, чтобы научиться эффективно устранять возмущения и восстанавливать исходные образцы.
- **Оптимизация**: Процесс обучения включает оптимизацию функций, которые измеряют разницу между восстановленными и исходными чистыми образцами, гарантируя, что процесс восстановления не изменит невредоносных характеристик образцов.

3) Показатели оценки

Эффективность процесса восстановления образца оценивается с использованием таких показателей, как точность восстановления, точность обнаружения и устойчивость к атакам противника.

4) Проблемы

- **Баланс между восстановлением и обнаружением**: Одна из проблем заключается в балансировке процесса восстановления таким образом, чтобы он эффективно устранял нежелательные модификации без снижения производительности системы обнаружения невредоносных образцов.
- **Вычислительные издержки**: Реализация точного восстановления приводит к дополнительным вычислительным затратам, которые необходимо сбалансировать с преимуществами повышения точности и надёжности обнаружения.

5) Преимущества

- **Улучшенная точность**: Точное восстановление образцов повышает точность обнаружения системой обнаружения вредоносных программ, гарантируя эффективное устранение вредоносных модификаций.
- **Повышенная надёжность**: благодаря точному восстановлению исходных образцов система становится более устойчивой к атакам противника, что делает её более устойчивой к тактикам уклонения, используемым злоумышленниками.

6) Интеграция с системами обнаружения

Восстановленные образцы передаются в существующие системы обнаружения вредоносных программ, которые затем могут точно классифицировать их.

IV. Эксперментальная часть

A. Основные результаты

1) Экспериментальная установка

- **Наборы данных**: В экспериментах используются несколько наборов данных приложений Android, включая как невредоносные (чистые), так и вредоносные образцы
- **Состязательные атаки**: для создания примеров используются различные методы состязательной атаки, включая атаки как с использованием белого ящика, так и с использованием черного ящика.
- **Показатели оценки**: Производительность MalPurifier оценивается с использованием таких показателей, как точность обнаружения, частота ложных срабатываний и устойчивость к атакам противника.

2) Базовые модели

В экспериментах сравнивается MalPurifier с несколькими базовыми моделями, включая традиционные системы обнаружения вредоносных программ на основе машинного обучения.

3) Результаты

- **Точность обнаружения**: MalPurifier значительно повышает точность обнаружения вредоносных программ для Android и показывает, что процесс очистки эффективно устраняет нежелательные модификации, позволяя модели обнаружения точно классифицировать образцы.
- **Частота ложноположительных результатов**: Частота ложноположительных результатов снижается при использовании MalPurifier, что указывает на то, что процесс очистки не создаёт значительных шумов, которые могли бы привести к неправильной классификации чистых образцов.
- **Надёжность**: MalPurifier повышает надёжность систем обнаружения вредоносных программ, делая их более устойчивыми к различным типам атак противника (белого, серого и чёрного ящика)

4) Исследование абляции

Проводится исследование абляции для оценки вклада каждого компонента системы MalPurifier. Исследование показывает, что каждый компонент, включая модуль очистки и систему подавления защитного шума, вносит свой вклад в общую эффективность системы.

5) Сравнение с базовыми показателями

Эксперименты демонстрируют, что MalPurifier превосходит базовые модели с точки зрения точности обнаружения и надёжности и демонстрирует производительность при очистке образцов и улучшении системы обнаружения вредоносных программ.

6) Тематические исследования

Представлены конкретные тематические исследования, иллюстрирующие эффективность средства для MalPurifier в реальных сценариях, показывающие как фреймворк может справляться с различными типами атак и улучшать обнаружение сложных образцов вредоносного ПО.

B. Эффективность и затраты без атак

Основная цель – оценить базовую производительность MalPurifier с точки зрения точности обнаружения и вычислительных затрат при отсутствии атак противника.

1) Экспериментальная установка

- **Наборы данных**: В экспериментах использовались стандартные наборы данных приложений Android, включая образцы как чистых, так и вредоносных программ, для оценки производительности.
- **Показатели оценки**: включают точность обнаружения, частоту ложноположительных результатов и вычислительные издержки.

2) Точность обнаружения

- **Производительность**: MalPurifier демонстрирует высокую точность обнаружения при отсутствии атак, а процесс очистки не снижает производительность системы обнаружения вредоносных программ на чистых образцах.
- **Сравнение с базовыми показателями**: Точность обнаружения MalPurifier сравнима с традиционными системами обнаружения вредоносных программ без очистки или даже выше.

3) Частота ложноположительных результатов

- **Оценка**: Частота ложноположительных результатов оценивается для гарантии, что процесс очистки не приведёт к неправильной классификации чистых образцов.
- **Результаты**: Частота ложноположительных результатов остаётся низкой, что указывает на то, что MalPurifier поддерживает высокую точность в различении чистых и вредоносных образцов.

4) Вычислительные затраты

- **Накладные расходы**: Вычислительные затраты на процесс очистки измеряются для оценки возможности развёртывания устройства в реальных сценариях с вредоносным кодом.
- **Результаты**: хотя процесс очистки сопряжён с некоторыми вычислительными затратами, он находится в приемлемых пределах для практического внедрения. Накладные расходы оправданы значительным повышением точности и надёжности обнаружения.

5) Вывод:

- **Эффективность**: MalPurifier эффективен для поддержания высокой точности обнаружения и низкого уровня ложноположительных результатов даже при отсутствии атак противника.
- **Стоимость**: Вычислительные затраты на процесс очистки являются управляемыми, что делает MalPurifier жизнеспособным решением для повышения надёжности систем обнаружения вредоносных программ Android.

C. Устойчивость к атакам с обфускацией

Основная цель - оценить, насколько хорошо MalPurifier может обрабатывать атаки с обфускацией, которые являются распространённым методом, используемым для обхода систем обнаружения вредоносных программ.

1) Методы обфускации

- **Типы обфускации**: рассматриваются различные методы обфускации, такие как шифрование кода, упаковка и полиморфизм, которые изменяют внешний вид вредоносного ПО без изменения его функциональности.
- **Состязательные образцы**: создаются с использованием этих методов обфускации для проверки надёжности MalPurifier.

2) Экспериментальная установка

- **Наборы данных**: В экспериментах используются наборы данных приложений Android, которые включают образцы скрытого вредоносного ПО.
- **Показатели**: Показатели оценки включают точность обнаружения, частоту ложноположительных результатов и устойчивость к атакам с обфускацией.

3) Точность обнаружения

- **Производительность**: MalPurifier демонстрирует высокую точность обнаружения даже при работе с обфусцированными образцами вредоносных программ. Процесс очистки эффективно устраняет путаницу, позволяя модели обнаружения точно классифицировать образцы.
- **Сравнение с базовыми показателями**: Точность обнаружения MalPurifier значительно выше, чем у традиционных систем обнаружения вредоносных программ, которые не используют очистку.

4) Частота ложноположительных результатов

- **Оценка**: Частота ложноположительных результатов оценивается для гарантии того, что в процессе очистки невредоносные образцы не будут ошибочно классифицированы как вредоносные из-за обфусцирования.
- **Результаты**: Частота ложноположительных результатов остаётся низкой, указывая, что MalPurifier поддерживает высокую точность в различении чистых и обфусцированных образцов.

5) Надёжность

- **Эффективность**: Результаты показывают, что MalPurifier устойчиво к различным методам обфускации. Процесс очистки успешно нейтрализует последствия обфускации, повышая общую надёжность системы обнаружения вредоносных программ.
- **Состязательное обучение**: Надёжность обусловлена процессом обучения, который включает в себя разнообразный набор методов обфускации, позволяющих модулю очистки справляться с различными типами обфускации.

6) Вывод

- **Повышенная безопасность**: MalPurifier значительно повышает безопасность систем обнаружения вредоносных программ Android, обеспечивая надёжную защиту от атак с обфускацией.
- **Практические последствия**: Результаты демонстрируют практическую применимость MalPurifier в реальных сценариях, где злоумышленники обычно используют обфускацию, чтобы избежать обнаружения.

D. Устойчивость к «атакам без знания»

Основная цель состоит в том, чтобы оценить устойчивость MalPurifier к незаметным атакам, когда злоумышленник не имеет представления о действующем защитном механизме.

1) Сценарий внезапной атаки:

- **Определение**: «Атаки без знания» — это те, при которых злоумышленник не знает об используемых конкретных защитных механизмах, не принимая во внимание наличие MalPurifier.
- **Методы атаки**: для создания примеров состязательности используются различные методы атаки, включая методы как "белого ящика", так и "черного ящика".

2) Экспериментальная установка

- **Наборы данных**: В экспериментах используются наборы данных приложений Android, включая как чистые, так и вредоносные образцы, для оценки производительности в сценариях незаметных атак.
- **Показатели**: Показатели оценки включают точность обнаружения, частоту ложноположительных срабатываний и устойчивость к атакам без предупреждения.

3) Точность обнаружения

- **Производительность**: MalPurifier демонстрирует высокую точность обнаружения даже при столкновении с примерами состязательности, созданными в сценариях забывчивой атаки. Процесс очистки эффективно устраняет нежелательные модификации, позволяя модели обнаружения точно классифицировать образцы.
- **Сравнение с базовыми показателями**: Точность обнаружения MalPurifier значительно выше, чем у традиционных систем обнаружения вредоносных программ, которые не используют очистку.

4) Частота ложноположительных результатов

- **Оценка**: Частота ложноположительных результатов оценивается для гарантии того, что в процессе очистки чистые образцы не будут ошибочно классифицированы как вредоносные.
- **Результаты**: Частота ложноположительных результатов остаётся низкой, указывая, что для MalPurifier сохраняет высокую точность при различении чистых образцов от вредоносных.

5) Надёжность

- **Эффективность**: Результаты показывают, что MalPurifier устойчиво к непроизвольным атакам. Процесс очистки успешно нейтрализует воздействие вредоносных модификаций, повышая общую надёжность системы обнаружения вредоносных программ.
- **Обучение**: Надёжность обусловлена процессом состязательного обучения, который включает в себя

разнообразный набор состязательных примеров, позволяющих модулю очистки научиться справляться с различными типами атак.

6) Вывод

- **Повышенная безопасность**: MalPurifier значительно повышает безопасность систем обнаружения вредоносных программ Android, обеспечивая надёжную защиту от незаметных атак.
- **Практические последствия**: Полученные результаты демонстрируют практическую применимость MalPurifier в реальных сценариях, когда злоумышленники могут не знать о конкретных действующих защитных механизмах.

E. Устойчивость к адаптивным атакам

Основная цель – оценить устойчивость MalPurifier к адаптивным атакам, когда злоумышленник осведомлён о защитном механизме и пытается его обойти.

1) Сценарий адаптивной атаки

- **Определение**: Адаптивные атаки — это те, при которых злоумышленник знает о защитном механизме (MalPurifier) и адаптирует свою стратегию атаки, чтобы обойти его.
- **Методы атаки**: для создания состязательных примеров используются различные сложные методы атаки, специально разработанные для обхода процесса очистки.

2) Экспериментальная установка

- **Наборы данных**: В экспериментах используются наборы данных приложений Android, включая как невредоносные (чистые), так и вредоносные образцы, для оценки производительности в сценариях адаптивных атак.
- **Показатели оценки**: включают точность обнаружения, частоту ложноположительных результатов и устойчивость к адаптивным атакам.

3) Точность обнаружения

- **Производительность**: MalPurifier демонстрирует высокую точность обнаружения даже при столкновении с примерами состязательности, созданными в сценариях адаптивной атаки. Процесс очистки эффективно устраняет нежелательные модификации, позволяя модели обнаружения точно классифицировать образцы.
- **Сравнение с базовыми показателями**: Точность обнаружения MalPurifier значительно выше, чем у традиционных систем обнаружения вредоносных программ, которые не используют очистку.

4) Частота ложноположительных результатов

- **Оценка**: Частота ложноположительных результатов оценивается для гарантии того, что в процессе очистки невредоносные (чистые) образцы не будут ошибочно классифицированы как вредоносные из-за неблагоприятных воздействий.
- **Результаты**: Частота ложноположительных результатов остаётся низкой, что указывает на то, что MalPurifier сохраняет высокую точность при различении невредоносных образцов от образцов с враждебными нарушениями.

5) Надёжность

- **Эффективность**: Результаты показывают, что MalPurifier устойчиво к адаптивным атакам. Процесс очистки успешно нейтрализует воздействие вредоносных модификаций, повышая общую надёжность системы обнаружения вредоносных программ.
- **Обучение**: Надёжность обусловлена процессом состязательного обучения, который включает в себя разнообразный набор состязательных примеров, позволяющих модулю очистки научиться справляться с различными типами атак.

6) Вывод

- **Повышенная безопасность**: MalPurifier значительно повышает безопасность систем обнаружения вредоносных программ Android, обеспечивая надёжную защиту от адаптивных атак.
- **Практические последствия**: Полученные результаты демонстрируют практическую применимость MalPurifier в реальных сценариях, где злоумышленники могут адаптировать свои стратегии для обхода защитных механизмов.