ПРОЕКТНАЯ ДЕЯТЕЛЬНОСТЬ

Оценка уязвимостей смарт-контрактов

Выполнила: Николаец Дарья 171-361

Куратор: Репин М.М.

ЦЕЛИ ПРОЕКТА

1. Анализ Blockchain-платформ и выбор наиболее оптимальной платформы для написания смарт-контрактов.

2. Выбор наиболее сбалансированной комбинации методики и инструмента анализа уязвимостей смарт-контрактов, которую можно использовать для поиска и анализа уязвимостей в коде смарт-контракта.

ЭТАПЫ РАБОТЫ НАД ПРОЕКТОМ

1. ВЫБОР НАИБОЛЕЕ ОПТИМАЛЬНОЙ ПЛАТФОРМЫ ДЛЯ РАБОТЫ СО СМАРТ-КОНТРАКТАМИ.

ЭТАП 1

Изучение рынка смарт-контрактов. Изучение принципа работы и случаев применения смарт-контрактов

ЭТАП 2

Анализ 4-5 Blockchain-платформ для написания смарт-контрактов, выделение , преимуществ и недостатков, а также описание характеристики

ЭТАП 3

Сравнение проанализированных Blockchain-платформ и выбор наиболее оптимальной

ЭТАПЫ РАБОТЫ НАД ПРОЕКТОМ

2.ВЫБОР НАИБОЛЕЕ СБАЛАНСИРОВАННОЙ КОМБИНАЦИЮ МЕТОДИКИ И ИНСТРУМЕНТА, ДЛЯ ПОИСКА И АНАЛИЗА УЯЗВИМОСТЕЙ В КОДЕ СМАРТ-КОНТРАКТА.

ЭТАП 1

Изучение принципа работы методов анализа уязвимостей смарт-контрактов (статический и динамический), анализ их преимуществ и недостатков

ЭТАП 2

Выбор и анализ методик и инструментов путем выделения их особенностей, преимуществ и недостатков

ЭТАП 3

Выбор наиболее сбалансированной комбинацию методики и инструмента, которые можно использовать для поиска и анализа уязвимостей в коде

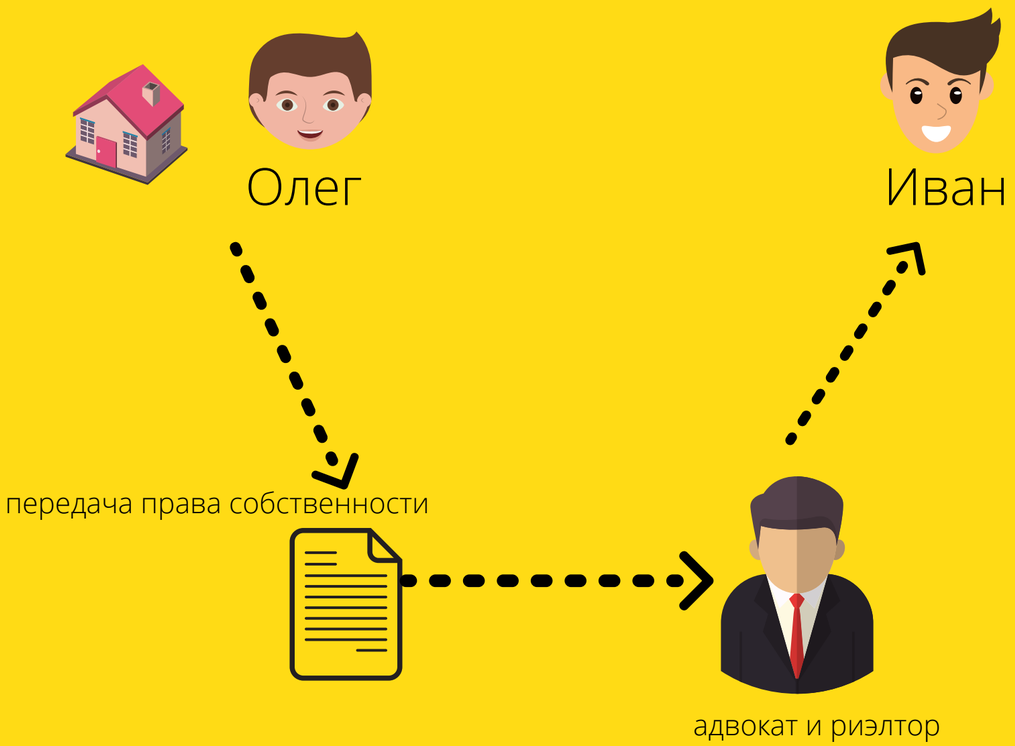
ЧТО ТАКОЕ СМАРТ-КОНТРАКТЫ?

Умный контракт (англ. Smart contracts) — электронный алгоритм, описывающий набор условий, выполнение которых влечет за собой некоторые события в реальном мире или цифровых системах. Для реализации умных контрактов требуется децентрализованная среда, полностью исключающая человеческий фактор, а для возможности использования в умном контракте передачи стоимости требуется криптовалюта.

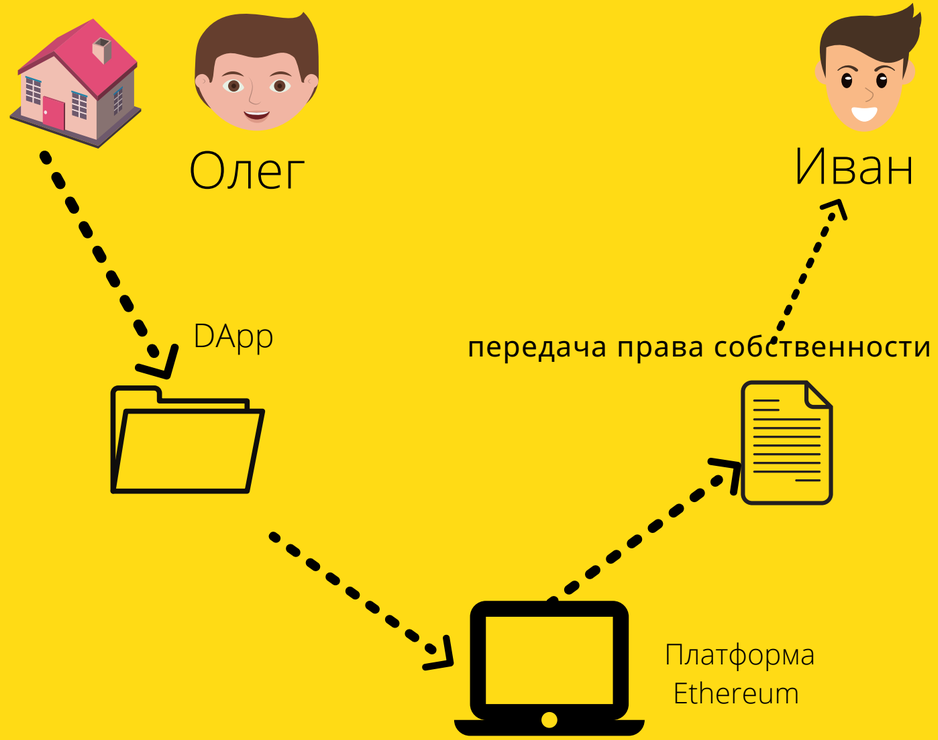
КАК РАБОТАЮТ СМАРТ-КОНТРАКТЫ

СИТУАЦИЯ: ИВАН ХОЧЕТ КУПИТЬ ДОМ У ОЛЕГА

Обычный контракт



Смарт-контракт



ПРЕИМУЩЕСТВА СМАРТ-КОНТРАКТОВ

Безопасно и без посредников

Автоматизация

Скорость

Надежность

Экономия

НЕДОСТАТКИ СМАРТ-КОНТРАКТОВ

Возможные уязвимости и ошибки

Нехватка специалистов

Невозможность изменения смарт-контрактов

Отсутствие регулирования

Сложность внедрения в реальные бизнес-процессы

ПЛАТФОРМЫ ДЛЯ СМАРТ-КОНТРАКОВ

NEO

ETHEREUM

WAVES

EOS

STELLAR

Ethereum

Ethereum является первой блокчейн платформой в которой возможно исполнение тьюринг-полных смарт-контрактов. Полнота по Тьюрингу в смарт-контрактах позволяет выполнить любую математически вычислительную функцию.

Ethereum завоевал популярность у пользователей благодаря возможности выпуску токенов на платформе Эфириума,созданию децентрализованных приложений а также весьма гибким возможностями по созданию смарт-контрактов.

Сфера применений смарт-контрактов в Ethereum чрезвычайно широка и любую логически поданную идею можно реализовать с помощью данной сети.

Stellar

Stellar
позволяет комфортно управлять микротранзакциями, в сети низкая комиссия и повышенная технологическая совместимость с финансовым рынком.

Блокчейн
Stellar поддерживает работу смарт-контрактов, однако они обладают ограничениями и сложный функционал на нём невозможен.

В
то же время на Stellar доступны мультиподписи, атомарность транзакций их последовательность и временные ограничения.

Waves

Её
отличительно особенностью является алгоритм консенсуса LPoS, что расшифровывается как Leased Proof of Stake или арендованное подтверждение доли.

В основной сети Waves сегодня работают
ограниченные по функционалу смарт-контракты которые позволяют выполнять
следующие сценарии: создание токена, заморозка токена, выплаты по расписанию,
двухфакторная аутентификация и много другое.

Waves - блокчейн платформа имеющая российское происхождение была основана в 2016 году.

NEO

Интересным отличием NEO является поддержка популярных языков программирования для разработки смарт-контрактов, например таких как Java, F#, C#, Kotlin, Go и Python.

NEO - блокчейн-платформа разрабатываемая сообществом китайских программистов

Ещё одной особенностью является использование отдельной криптовалюты GAS для оплаты транзакций и выполнения смарт-контрактов.

EOS

EOS на сегодняшний день является второй по популярность блокчейн платформой по применению смарт-контрактов и первой по активному числу транзакций.

Отличительной особенностью EOS можно назвать масштабируемость, параллельные вычисления и высокую пропускную способность сети.

В EOS используется алгоритм консенсуса Delegated Proof-of-Stake что повышает пропускную способность сети, но плохо сказывается на её децентрализации.

NEO

ОСОБЕННОСТИ

Алгоритм консенсуса dBFT, высокая скорость транзакций, централизация, SDK +

ЯЗЫКИ ПРОГРАММИРОВАНИЯ

C#, F#Java, KotlinGo, Python

ТРАНЗАКЦИЙ В СЕКУНДУ

1000 p/s

CONSENSUS MECHANISM

dBFT

ЦЕНА

0 $ для своей валюты и 0.001 $ для других

BLOCK TIME

15 sec

EOS

ОСОБЕННОСТИ

Высокая масштабируемость, высокая скорость транзакций, сложность для обывателя, SDK +

ЯЗЫКИ ПРОГРАММИРОВАНИЯ

C, C++, Байт-код WASM

ТРАНЗАКЦИЙ В СЕКУНДУ

4000 p/s

CONSENSUS MECHANISM

DPos

ЦЕНА

0.03 $

BLOCK TIME

0.5 sec

ETHEREUM

ОСОБЕННОСТИ

Высокая популярность, полнота по Тьюрингу, понятно для пользователя, SDK +

ЯЗЫКИ ПРОГРАММИРОВАНИЯ

Solidity, Serpent, Mutan

ТРАНЗАКЦИЙ В СЕКУНДУ

12-25 p/s

CONSENSUS MECHANISM

PoW

ЦЕНА

3-4 цента

BLOCK TIME

15 sec

STELLAR

ОСОБЕННОСТИ

Ограниченный функционал, высокая скорость, микроплатежи, SDK +

ЯЗЫКИ ПРОГРАММИРОВАНИЯ

JavaScript, Golang, Python

ТРАНЗАКЦИЙ В СЕКУНДУ

2000 p/s

CONSENSUS MECHANISM

SCP

ЦЕНА

1 цент для 10000 транзакций

BLOCK TIME

5 sec

WAWES

ОСОБЕННОСТИ

Алгоритм консенсуса LPoS, приватный блокчейн Vostok, SDK +

ЯЗЫКИ ПРОГРАММИРОВАНИЯ

RIDE, RIDEON

ТРАНЗАКЦИЙ В СЕКУНДУ

400 p/s

CONSENSUS MECHANISM

LPos

ЦЕНА

0.001-0.009 $

BLOCK TIME

3 sec

NEO

ПРЕИМУЩЕСТВА

Алгоритм консенсуса dBFT
Высокая скорость транзакций
Поддержка многих языков программирования, расширяющая перспективы коммерческого применения платформы.

НЕДОСТАТКИ

Разработчики могут самостоятельно
влиять на сеть и её участников – замораживать счета, следить за операциями,
предоставлять данные властям по их требованию.
Отсутствие анонимности.
Централизация

STELLAR

ПРЕИМУЩЕСТВА

Высокая скорость
Микроплатежи

НЕДОСТАТКИ

Не очень подходит для разработки более сложных смарт-контрактов

Ограниченный функционал

ETHEREUM

ПРЕИМУЩЕСТВА

Широкое распространение
Гибкость
Низкий порог вхождения
Четкие рекомендации для разработчиков
Много литературы / справок доступно

НЕДОСТАТКИ

Оплата транзакций в сети (gas)
Нагрузка на сеть
Проблемы с безопасностью
Проблемы масштабируемости
Относительно медленное подтверждение транзакции
Дороже,чем другие платформы

WAWES

ПРЕИМУЩЕСТВА

Высокая скорость
Имеет сильные стратегические партнерства
Имеет функционирующую децентрализованную биржу
Алгоритм консенсуса LPoS
Приватный блокчейн Vostok

НЕДОСТАТКИ

Есть некоторые проблемы в области безопасности

EOS

ПРЕИМУЩЕСТВА

Высокая масштабируемость
Высокая скорость транзакций

НЕДОСТАТКИ

Отсутствие главной сети EOS.

НАИБОЛЕЕ ОПТИМАЛЬНАЯ ПЛАТФОРМА ДЛЯ РАБОТЫ СО СМАРТ-КОНТРАКТАМИ

EOS

EOS ЯВЛЯЕТСЯ ТЕХНОЛОГИЧЕСКИ ИНТЕРЕСНОЙ И ПОТЕНЦИАЛЬНО ВОСТРЕБОВАННОЙ В ШИРОКОМ ДИАПАЗОНЕ ПРИМЕНЕНИЙ СМАРТ-КОНТРАКТОВ. ОТЛИЧИТЕЛЬНОЙ ОСОБЕННОСТЬЮ EOS МОЖНО НАЗВАТЬ МАСШТАБИРУЕМОСТЬ,ПАРАЛЛЕЛЬНЫЕ ВЫЧИСЛЕНИЯ И ВЫСОКУЮ ПРОПУСКНУЮ СПОСОБНОСТЬ СЕТИ.

РАССМОТРИМ ДВА НАПРАВЛЕНИЯ
В АНАЛИЗЕ
БЕЗОПАСНОСТИ УЯЗВИМОСТЕЙ.

Статический и динамический анализ.

СТАТИЧЕСКИЙ АНАЛИЗ

Статический анализ кода – это процесс выявления ошибок и недочетов в исходном коде программ. Статический анализ можно рассматривать как автоматизированный процесс обзора кода (code review).

ДИНАМИЧЕСКИЙ АНАЛИЗ

Динамический анализ кода – это способ анализа программы непосредственно при ее выполнении.

ДИНАМИЧЕСКИЙ АНАЛИЗ

ПРЕИМУЩЕСТВА

Работает в терминах конкретных ячеек памяти, что позволяет проводить анализ даже при интенсивном использовании указателей.
Относительная независимость от платформы, фреймворков и языков, на которых разработано приложение.
Ложные срабатывания почти исключены.

НЕДОСТАТКИ

Невысокая степень покрытия. Далеко не все вызовы API и точки входа можно легко обнаружить.
Драматическое падение эффективности при усложнении клиента/протокола.
Долгое время работы.
Сложность выявления многих типов. Например, ошибки использования криптографии, такие как слабые механизмы генерации cookie или session ID.

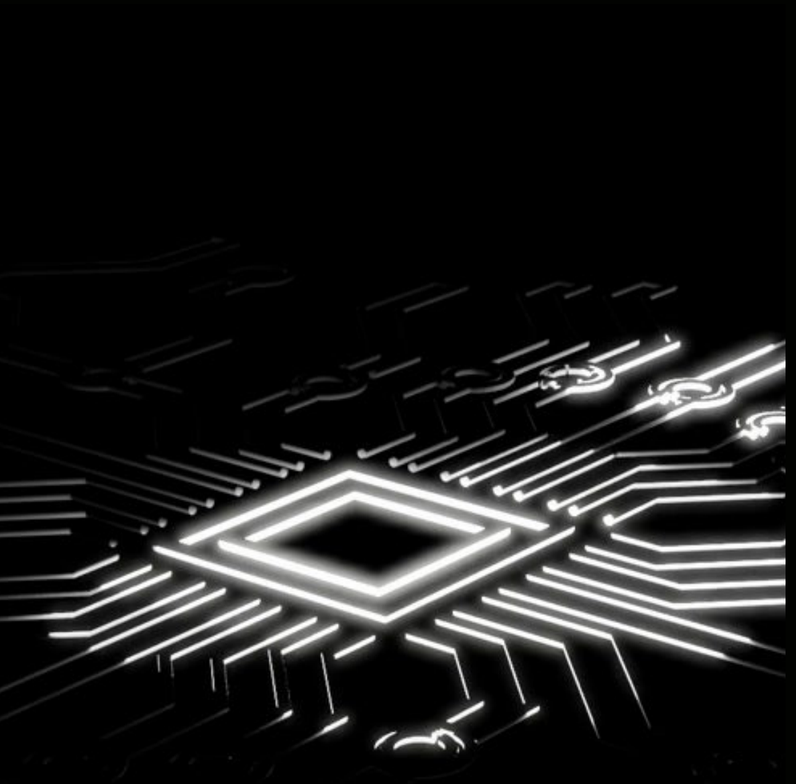
СТАТИЧЕСКИЙ АНАЛИЗ

ПРЕИМУЩЕСТВА

Полное покрытие анализируемого кода.
Нет необходимости выполнять приложение в боевой среде. Статический анализ можно внедрять на самых ранних стадиях разработки, минимизируя стоимость найденных уязвимостей.

НЕДОСТАТКИ

Неизбежное наличие ложных срабатываний, потребление ресурсов и длительное время сканирований на больших объемах кода. Однако, эти минусы неизбежны, исходя из специфики алгоритмов.



ОБЗОР МЕТОДИК АНАЛИЗА УЯЗВИМОСТЕЙ СМАРТ-КОНТРАКТОВ.

CODE INSTRUMENTATION
SYMBOLIC EXECUTION
CONSTRAINT SOLVING
ABSTRACT INTERPRETATION
HORN LOGIC
MODEL CHECKING

HORN LOGIC

ЭТО ОГРАНИЧЕННАЯ ФОРМА ЛОГИКИ ПЕРВОГО ПОРЯДКА, ГДЕ ВСЕ ФОРМУЛЫ (ПРЕДЛОЖЕНИЯ) ЯВЛЯЮТСЯ ПРАВИЛАМИ IF–THEN. ХОТЯ ЛОГИКА ХОРНА И ОГРАНИЧЕНА, ОНА ВСЕ ЖЕ ЯВЛЯЕТСЯ ВЫЧИСЛИТЕЛЬНО УНИВЕРСАЛЬНОЙ, ПОЭТОМУ МОЖЕТ ВЫПОЛНЯТЬ ТЕ ЖЕ ВЫЧИСЛЕНИЯ, ЧТО И ЛЮБОЙ КОМПЬЮТЕР.

MODEL CHECKING

ЭТО МЕТОД АВТОМАТИЧЕСКОЙ ПРОВЕРКИ ПРАВИЛЬНОСТИ СВОЙСТВ КОНЕЧНЫХ СИСТЕМ. ДЛЯ ЭТОГО ТРЕБУЕТСЯ МОДЕЛЬ СИСТЕМЫ, КОТОРАЯ ЗАТЕМ ПРОВЕРЯЕТСЯ НА СООТВЕТСТВИЕ ЗАДАННОЙ СПЕЦИФИКАЦИИ.

ABSTRACT INTERPRETATION

ИГНОРИРУЕТ ОПРЕДЕЛЕННЫЕ ИНСТРУКЦИИ ИЛИ ОПРЕДЕЛЕННЫЕ ЭФФЕКТЫ ИНСТРУКЦИЙ ПРИ ВЫПОЛНЕНИИ БАЙТ-КОДА. ЭТО МОЖНО СДЕЛАТЬ, ПЕРЕВЕДЯ ИНСТРУКЦИИ В ДРУГОЙ ФОРМАЛИЗМ, НАПРИМЕР DATALOG, А ЗАТЕМ ИЗУЧИВ ВСЕ ВОЗМОЖНЫЕ ВАРИАНТЫ ВЫПОЛНЕНИЯ.

CONSTRAINT SOLVING

ОЗНАЧАЕТ ОПРЕДЕЛЕНИЕ РАЗРЕШИМОСТИ ОГРАНИЧЕНИЙ И ВОЗМОЖНОСТЬ ВЫЧИСЛЕНИЯ КОНКРЕТНОГО РЕШЕНИЯ. ОГРАНИЧЕНИЕ – ЭТО НАБОР УСЛОВИЙ, КОТОРЫМ ДОЛЖНЫ УДОВЛЕТВОРЯТЬ ПЕРЕМЕННЫЕ. В НАШЕМ КОНТЕКСТЕ ОГРАНИЧЕНИЯ В ОСНОВНОМ ВОЗНИКАЮТ ИЗ УСЛОВИЙ ВЕТВЛЕНИЯ В КОДЕ.

SYMBOLIC EXECUTION

ЭТО СРЕДСТВО АНАЛИЗА ПРОГРАММЫ ДЛЯ ОПРЕДЕЛЕНИЯ ТОГО, КАКИЕ ВХОДНЫЕ ДАННЫЕ ВЫЗЫВАЮТ ВЫПОЛНЕНИЕ КАКОЙ ЧАСТИ ПРОГРАММЫ. ИНТЕРПРЕТАТОР СЛЕДУЕТ ЗА ПРОГРАММОЙ, ПРИНИМАЯ СИМВОЛИЧЕСКИЕ ЗНАЧЕНИЯ ДЛЯ ВХОДНЫХ ДАННЫХ, А НЕ ПОЛУЧАЯ ФАКТИЧЕСКИЕ ВХОДНЫЕ ДАННЫЕ, КАК ЭТО БЫЛО БЫ ПРИ ОБЫЧНОМ ВЫПОЛНЕНИИ ПРОГРАММЫ. ДРУГИМИ СЛОВАМИ, ОНО ПОЗВОЛЯЕТ НАХОДИТЬ НЕДОЧЕТЫ В КОДЕ, ДАЖЕ НЕ ЗНАЯ, КАКОЕ ЗНАЧЕНИЕ ПЕРЕМЕННЫХ БУДЕТ В СТРОКЕ С ОШИБКОЙ.

CODE INSTRUMENTATION

ЭТО ОТСЛЕЖИВАНИЕ ПАРАМЕТРОВ УРОВНЯ ПРОИЗВОДИТЕЛЬНОСТИ КОДА, ВОЗМОЖНОСТЬ ДИАГНОСТИРОВАТЬ ОШИБКИ И ЗАПИСЫВАТЬ ИНФОРМАЦИЮ НА ВСЁМ ПРОТЯЖЕНИИ РАБОТЫ ДЛЯ ОТСЛЕЖИВАНИЯ ПРИЧИН ИХ ВОЗНИКНОВЕНИЯ.
ТАКИМ ОБРАЗОМ МОЖНО ПРОВЕРИТЬ СМАРТ-КОНТРАКТ В «БОЕВЫХ» УСЛОВИЯХ.

НАИБОЛЕЕ СБАЛАНСИРОВАННАЯ МЕТОДИКА ДЛЯ АНАЛИЗА УЯЗВИМОСТЕЙ СМАРТ-КОНТРАКТОВ - SYMBOLIC EXECUTION

Есть много академических проектов, которые оказали большое влияние на реальный мир, например, на обнаружение важных ошибок в программном обеспечении с открытым исходным кодом, с помощью символьного выполнения.

При исследовании проблемы символьное выполнение может создать входные данные и трассировку, путь выполнения, которые можно запустить в реальной программе и выполнить эту программу на основе этих входных данных. И после этого мы можете выявить реальный баг и приступить к его исправлению, используя традиционные механизмы отладки. И это особенно ценно, когда вы находитесь в промышленной среде разработки, где у вас, вероятно, нет времени, чтобы заботиться о каждой маленькой проблеме в вашем коде.

ОБЗОР ИНСТРУМЕНТОВ ДЛЯ АНАЛИЗА УЯЗВИМОСТЕЙ СМАРТ-КОНТРАКТОВ.

Scompile, Mythril, Securify, Manticore, MAIAN.

SCOMPILE

SCOMPILE БЕРЕТ БАЙТ-КОД КОНТРАКТА, СТРОИТ **CFG**, ОПРЕДЕЛЯЕТ ВСЕ ВЫЧИСЛИТЕЛЬНЫЕ ПУТИ, ВКЛЮЧАЮЩИЕ ЛЮБОЙ ПОТОК ЭФИРА, ВЫБИРАЕТ ТЕ, КОТОРЫЕ СООТВЕТСТВУЮТ ПАТТЕРНАМ, ХАРАКТЕРНЫМ ДЛЯ ОПРЕДЕЛЕННЫХ УЯЗВИМОСТЕЙ, РАНЖИРУЕТ ИХ ЭВРИСТИЧЕСКИ В СООТВЕТСТВИИ С РЕЛЕВАНТНОСТЬЮ И, НАКОНЕЦ, ПРИМЕНЯЕТ СИМВОЛИЧЕСКОЕ ИСПОЛНЕНИЕ, ПРЕЖДЕ ЧЕМ ПРЕДСТАВИТЬ РЕЗУЛЬТАТ ПОЛЬЗОВАТЕЛЮ ДЛЯ РУЧНОЙ ПРОВЕРКИ.

MYTHRIL

ЭТО ИНСТРУМЕНТ КОМАНДНОЙ СТРОКИ В **PYTHON** ДЛЯ ИНТЕРАКТИВНОГО АНАЛИЗА СМАРТ-КОНТРАКТОВ. ОН ВЫПОЛНЯЕТ БАЙТ-КОД ВИРТУАЛЬНОЙ МАШИНЫ СИМВОЛИЧЕСКИ И ВИЗУАЛИЗИРУЕТ **CFG**, ПРИЧЕМ УЗЛЫ, СОДЕРЖАЩИЕ РАЗОБРАННЫЙ КОД, И РЕБРА ПОМЕЧАЮТСЯ ФОРМУЛАМИ ПУТИ. ПРОВЕРЕННЫЕ УЯЗВИМОСТИ ПОДРОБНО ОПИСАНЫ В ИНТЕРАКТИВНОЙ ДОКУМЕНТАЦИИ. **MYTHRIL** РАЗРАБАТЫВАЕТСЯ И ПОДДЕРЖИВАЕТСЯ КОМПАНИЕЙ **CONSENSUS**, А ТАКЖЕ ДОСТУПЕН НА **GITHUB** ПОД ЛИЦЕНЗИЕЙ **MIT** С СЕНТЯБРЯ **2017** ГОДА.

SECURIFY

ПРИНИМАЕТ БАЙТ-КОД EVM И СВОЙСТВА БЕЗОПАСНОСТИ В КАЧЕСТВЕ ВХОДНЫХ ДАННЫХ. ИНСТРУМЕНТ ДЕКОМПИЛИРУЕТ БАЙТ-КОД, ОРИЕНТИРОВАННЫЙ НА СТЕК, В ФОРМУ, ОСНОВАННУЮ НА ПРИСВОЕНИИ, И ПРЕДСТАВЛЯЕТ КОД В ВИДЕ ФАКТОВ ЖУРНАЛА ДАННЫХ. ЗАТЕМ ОН ВЫВОДИТ ДОПОЛНИТЕЛЬНЫЕ ФАКТЫ, КОТОРЫЕ ОПИСЫВАЮТ УПРАВЛЕНИЕ И ПОТОК ДАННЫХ В АБСТРАКТНОЙ ФОРМЕ.
ЭТОТ ИНСТРУМЕНТ НАПИСАН НА JAVA И ДОСТУПЕН НА **GITHUB** ПОД ЛИЦЕНЗИЕЙ **APACHE 2.0** С СЕНТЯБРЯ **2018** ГОДА. КРОМЕ ТОГО, ДОСТУП К ЗАКРЫТОЙ ВЕРСИИ ИСХОДНОГО КОДА МОЖНО ПОЛУЧИТЬ ЧЕРЕЗ ВЕБ-САЙТ КОМПАНИИ **CHAIN SECURITY**.

MANTICORE

ИСПОЛЬЗУЕТ СИМВОЛИЧЕСКОЕ ВЫПОЛНЕНИЕ, ЧТОБЫ НАЙТИ УНИКАЛЬНЫЕ ПУТИ ВЫЧИСЛЕНИЙ В ЭВМ И ДВОИЧНЫЕ ELF-ФАЙЛЫ. ОН ЗАПИСЫВАЕТ СООТВЕТСТВУЮЩИЕ СЛЕДЫ ВЫПОЛНЕНИЯ. ЧТО КАСАЕТСЯ ЭВМ, ТО МАНТИКОРА КОМПИЛИРУЕТ КОД СОЛИДНОСТИ В БАЙТ-КОД ДЛЯ ЕГО АНАЛИЗА, ПРОВЕРЯЕТ ТРАССИРОВКИ НА НАЛИЧИЕ УЯЗВИМОСТЕЙ, ТАКИХ КАК ПОВТОРНОЕ ПРОНИКНОВЕНИЕ И ДОСТИЖИМЫЕ ОПЕРАЦИИ САМОРАЗРУШЕНИЯ, И СООБЩАЕТ О НИХ В КОНТЕКСТЕ ИСХОДНОГО КОДА. ЭТОТ ИНСТРУМЕНТ РАЗРАБОТАН И ПОДДЕРЖИВАЕТСЯ КОМПАНИЕЙ **TRAIL OF BITS** И ДОСТУПЕН НА **GITHUB** ПОД ЛИЦЕНЗИЕЙ **AGPL-3.0** С ФЕВРАЛЯ **2017** ГОДА. ЕГО МОЖНО ИСПОЛЬЗОВАТЬ ИЗ КОМАНДНОЙ СТРОКИ ИЛИ ЧЕРЕЗ **API PYTHON**.

MAIAN

ИСПОЛЬЗУЕТ СИМВОЛИЧЕСКОЕ ВЫПОЛНЕНИЕ, ЧТОБЫ НАЙТИ УНИКАЛЬНЫЕ ПУТИ ВЫЧИСЛЕНИЙ В ЭВМ И ДВОИЧНЫЕ ELF-ФАЙЛЫ. ОН ЗАПИСЫВАЕТ СООТВЕТСТВУЮЩИЕ СЛЕДЫ ВЫПОЛНЕНИЯ. ЧТО КАСАЕТСЯ ЭВМ, ТО МАНТИКОРА КОМПИЛИРУЕТ КОД СОЛИДНОСТИ В БАЙТ-КОД ДЛЯ ЕГО АНАЛИЗА, ПРОВЕРЯЕТ ТРАССИРОВКИ НА НАЛИЧИЕ УЯЗВИМОСТЕЙ, ТАКИХ КАК ПОВТОРНОЕ ПРОНИКНОВЕНИЕ И ДОСТИЖИМЫЕ ОПЕРАЦИИ САМОРАЗРУШЕНИЯ, И СООБЩАЕТ О НИХ В КОНТЕКСТЕ ИСХОДНОГО КОДА. ЭТОТ ИНСТРУМЕНТ РАЗРАБОТАН И ПОДДЕРЖИВАЕТСЯ КОМПАНИЕЙ **TRAIL OF BITS** И ДОСТУПЕН НА **GITHUB** ПОД ЛИЦЕНЗИЕЙ **AGPL-3.0** С ФЕВРАЛЯ **2017** ГОДА. ЕГО МОЖНО ИСПОЛЬЗОВАТЬ ИЗ КОМАНДНОЙ СТРОКИ ИЛИ ЧЕРЕЗ **API PYTHON**.

ОБЗОР ИНСТРУМЕНТОВ, УКАЗЫВАЮЩИХ НАЗНАЧЕНИЕ, УРОВЕНЬ КОДА, ТИП, ПРЕДВАРИТЕЛЬНУЮ ОБРАБОТКУ И МЕТОДЫ АНАЛИЗА.

	Инструменты				
Назначение, уровень кода, тип, предварительная обработка, методы анализа	SCompile	Manticore	Mythril	Securify	MAIAN
Проблемы с безопасностью	✓	✓	✓	✓	✓
Эксплоиты		✓	✓		✓
Общий анализ	✓			✓	✓
Байткод		✓	✓	✓	✓
Solidity code	✓				
Статический анализ	✓	✓	✓	✓	✓
Динамический анализ					✓
Contextualization	✓	✓	✓	✓	
Disassembly	✓	✓	✓	✓	✓
Control flow graph	✓		✓		✓
Decompilation				✓	
Code instrumentation					
Symbolic execution	✓	✓	✓		✓
Constraint solving	✓	✓	✓		✓
Abstract interpretation				✓	
Horn Logic				✓	
Model checking					

ДЛЯ МЕТОДИКИ SYMBOLIC EXECUTION ЛУЧШИМ ИНСТРУМЕНТОМ ЯВЛЯЕТСЯ MAIAN.

Maian проверяет наличие трех типов контрактов на баги:

- Самоубийственные контракты (могут быть убиты кем угодно, например, контракт с библиотекой Parity Wallet);
- Блудные контракты (может отправить эфир кому угодно);
- Жадные контракты (никто не может выйти из эфира).

Maian анализирует смарт-контракты, определенные в файле с помощью:

- Солидность исходного кода;
- Источник байт-кода;

MAIAN x SYMBOLIC EXECUTION

НАИБОЛЕЕ СБАЛАНСИРОВАННАЯ КОМБИНАЦИЯ МЕТОДИКИ И ИНСТРУМЕНТА, КОТОРУЮ МОЖНО ИСПОЛЬЗОВАТЬ ДЛЯ ПОИСКА И АНАЛИЗА УЯЗВИМОСТЕЙ В КОДЕ СМАРТ-КОНТРАКТА.

РЕЗУЛЬТАТЫ

В ХОДЕ ПРОЕКТНОЙ ДЕЯТЕЛЬНОСТИ БЫЛИ РАССМОТРЕНЫ САМЫЕ ИННОВАЦИОННЫЕ, ОРИГИНАЛЬНЫЕ И УДОБНЫЕ ПЛАТФОРМЫ, ВЫДЕЛЕНЫ ИХ ХАРАКТЕРИСТИКИ, ПРЕИМУЩЕСТВА И НЕДОСТАТКИ, А ТАКЖЕ ВЫБРАНА НАИБОЛЕЕ СБАЛАНСИРОВАННАЯ ПЛАТФОРМА, КОТОРУЮ ЦЕЛЕСООБРАЗНО ИСПОЛЬЗОВАТЬ ДЛЯ РАБОТЫ СО СМАРТ-КОНТРАКТАМИ. ДАННАЯ РАБОТА ПОМОЖЕТ С ВЫБОРОМ НЕ ТОЛЬКО РАЗРАБОТЧИКАМ, НО И ОБЫЧНЫМ БИЗНЕСАМ В РАБОТЕ СО СМАРТ-КОНТРАКТАМИ.

ТАКЖЕ БЫЛО ПРОВЕДЕНО ИССЛЕДОВАНИЕ, КОТОРОЕ ПОКАЗЫВАЕТ, КАКУЮ НАИБОЛЕЕ СБАЛАНСИРОВАННУЮ КОМБИНАЦИЮ МЕТОДИКИ И ИНСТРУМЕНТА, МОЖНО ИСПОЛЬЗОВАТЬ ДЛЯ ПОИСКА И АНАЛИЗА УЯЗВИМОСТЕЙ В КОДЕ СМАРТ-КОНТРАКТА.ИССЛЕДОВАНИЕ ПРЕДНАЗНАЧЕНО ДЛЯ ТЕХ, КТО НАМЕРЕН АНАЛИЗИРОВАТЬ УЖЕ РАЗВЕРНУТЫЙ КОД, ХОЧЕТ РАЗРАБОТАТЬ БЕЗОПАСНЫЕ СМАРТ-КОНТРАКТЫ.