

Применение SAT-решателей для анализа стойкости криптографических хеш-функций

Давыдов Вадим Валерьевич
к.т.н., Университет ИТМО, ГУАП

Заикин Олег Сергеевич
к.т.н., ИДСТУ им. В.М. Матросова СО РАН

Кириянова Анастасия Павловна
Университет ИТМО

Москва, 2024

SAT (boolean satisfiability problem) –
задача
определения
выполнимости
булевой формулы в
конъюнктивной
нормальной форме

Работа 1971 года.
В работе
представлено
доказательство,
что задача SAT
является NP-
полной



Работа 1973 года.
Доказательство
получено
независимо от
Стивена Кука



Экземпляр SAT-задачи – булева формула, состоящая только из имён переменных, скобок, операций И, ИЛИ, НЕ

Задача:

Можно ли назначить всем переменным, встречающимся в формуле, значения ИСТИНА и ЛОЖЬ таким образом, чтобы формула стала истинной?

Пример

Выполнима ли формула:

$$\mathcal{F} = (x_1 \vee x_2) \wedge (x_1 \vee \neg x_2) \wedge (\neg x_1 \vee \neg x_2)$$

Это экземпляр задачи 2-SAT, который имеет полиномиальное решение:

$$\mathcal{F} = (x_1 \vee x_2) \wedge (x_1 \vee \neg x_2) \wedge (\neg x_1 \vee \neg x_2)$$

Выполняющий набор: x_1 - истина, x_2 - ложь

2-SAT

Определение выполнимости
КНФ, в которой в каждом
дизъюнкте ровно два
литерала

Принадлежит классу P

3-SAT

Определение выполнимости
КНФ, в которой в каждом
дизъюнкте ровно три
литерала

NP-полная задача

(любую КНФ можно за
полиномиальное время свести
к 3-SAT)

Основные полные SAT- алгоритмы

1962

Алгоритм DPLL – полный алгоритм для поиска с возвратом для решения задачи SAT-CNF

Выполняется путём выбора литерала и присвоения ему истинного значения, упрощения формулы и последующей рекурсивной проверки выполнимости формулы

Основные полные SAT- алгоритмы

1995

Алгоритм lookahead –
расширение DPLL

Основными отличиями от DPLL
являются эвристика выбора
переменных ветвления и
логический вывод failed literal
elimination

Основные полные SAT- алгоритмы

1996

Алгоритм **CDCL** (conflict-driven clause learning) – нехронологическое расширение алгоритма DPLL с памятью

В случае конфликта находится причина и осуществляется откат на несколько уровней. Чтобы запретить конфликт, формируется конфликтный дизъюнкт и добавляется к КНФ

Основные полные SAT- алгоритмы

Алгоритм **Cube-and-Conquer** использует в своей основе алгоритмы lookahead и CDCL

Сначала выполняется модифицированный алгоритм lookahead, где некоторым образом формируется куб – набор литералов

Для каждого куба формируется формула и решается с помощью CDCL-решателя

2011

SAT-решатели

SAT-решатель – программа, предназначенная для поиска выполняющего набора для функции в КНФ

Современные SAT-решатели – это сложные программные артефакты, включающие большое количество эвристик и оптимизаций для эффективной работы

SAT- решатели



Основаны на
одном или
нескольких
представленных
алгоритмах



Каждый год
проходит
соревнование
SAT-решателей



В 2019 году победу
одержал российский SAT-
решатель
MapleLCMDistChronoBT-DL



Сегодня одним
из лучших
решателей
является kissat

Хеш-функции

Криптографическая бесключевая хеш-функция h должна обладать следующими свойствами:

01 Стойкость к нахождению прообраза

Для любого хеша y вычислительно трудно найти любое сообщение m , такое что $h(m) = y$

02 Стойкость к нахождению второго прообраза

Для любого сообщения m вычислительно трудно найти m' , такое что $m' \neq m$ и $h(m) = h(m')$

03 Стойкость к нахождению коллизий

Вычислительно трудно найти два сообщения m и m' , такие что $m \neq m'$ и $h(m) = h(m')$

Конкурс NIST

2007 год

Объявлен конкурс SHA-3 на новую хеш-функцию

14/51

Хеш-функций вышли во второй раунд

2012 год





Конкурс завершён, победителем стал **КЕССАК**

Финалистами конкурса стали:

BLAKE, GRØSTL, SKEIN, JH и КЕССАК

SAT-криптоанализ криптографических хеш-функций

Порядок действий:

-  Выделить функцию сжатия
-  Преобразовать её к КНФ
-  Подставить в КНФ известную информацию
-  Попробовать найти выполняющий набор с помощью SAT-решателя

Анализ стойкости хеш-функций

01

Формируется единый файл с расширением .c
Определяется, что является входом и выходом функции
Из исходников убирается весь ненужный функционал

02

По исходнику формируется КНФ, которая подаётся на решатель
Проверяется корректность кодировки – при правильности реализации выполняющий набор находится моментально

03

Формируется КНФ, в которой выход фиксирован, а вход неизвестен
Это задача обращения дискретной функции, соответствующая задаче поиска прообраза, за разумное время решение не найти

Анализ стойкости хеш-функций

04

Задача обращения ослабляется путём сокращения числа раундов или шагов функции сжатия до получения решения

05

Проверяется корректность найденного прообраза

Необходимые программные средства



Программные комплексы для сведения задач криптоанализа к SAT:

- Sage
- SAW
- CBMC
- Transalg
- ...

SAT-решатели:

- Kissat
- Maple
- CryptoMiniSat
- TabularaSAT
- ...

Для анализа использовался ПК CBMC и SAT-решатель Kissat

Предыдущие результаты

В работе ^[1] были проанализированы финалисты конкурса NIST

- ⇒ **BLAKE** : обращено раундов – 1/14
- ⇒ **GRØSTL** : обращено раундов – 0.5/10
- ⇒ **KECCAK** : обращено раундов – 2/24
- ⇒ **JH** : обращено раундов – 2/42
- ⇒ **SKEIN** : обращено раундов – 6/72

[1] Homsirikamol E. et al. Security margin evaluation of SHA-3 contest finalists through SAT-based attacks //Computer Information Systems and Industrial Management: 11th IFIP TC 8 International Conference, CISIM 2012, Venice, Italy, September 26-28, 2012. Proceedings 11. – Springer Berlin Heidelberg, 2012. – С. 56-67.

Основные результаты

Анализировались 256-битные версии всех финалистов

BLAKE

- Анализовалась функция сжатия
- Задача поиска прообраза
- Обращено раундов:
1.1875 / 14

GRØSTL

- Анализовалась функция сжатия
- Задача поиска прообраза
- Обращено раундов:
1.625 / 10

КЕССАК

- Анализовалась хеш-функция целиком
- Задача поиска прообраза
- Обращено раундов:
1 / 24

Основные результаты

Анализировались 256-битные версии всех финалистов

JH

- Анализовалась функция сжатия
- Задача поиска прообраза
- Обращено раундов:

4 / 42

SKEIN

- Анализовалась функция сжатия
- Задача поиска псевдопрообраза
- Обращено раундов:

7 / 72

Направления дальнейших исследований



Комбинирование дифференциального и логического криптоанализа



Попытки упрощения отдельных операций, вызывающих резкие скачки сложности



Поиск коллизий с помощью SAT



Анализ отдельных структур с помощью SAT

Спасибо за внимание! Вопросы?

Контакты:

Давыдов Вадим Валерьевич

 vadimdavydov@outlook.com

 @vadimdavydov