Министерство науки и высшего образования Российской Федерации
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)
Институт прикладной математики и компьютерных наук
Кафедра компьютерной безопасности

ДОПУСТИТЬ К ЗАЩИТЕ В ГЭК
Руководитель ООП
канд. техн. наук, доцент
_____ С. А. Останин
“ 26 ” января 2021г.

ДИПЛОМНАЯ РАБОТА

УСЛОВИЯ КРИПТОАНАЛИТИЧЕСКОЙ ОБРАТИМОСТИ ФУНКЦИЙ

по основной образовательной программе подготовки специалистов
«Анализ безопасности компьютерных систем»
специальности 10.05.01 Компьютерная безопасность

Бердникова Наталья Юрьевна

Руководитель ВКР
заведующая лабораторией
компьютерной криптографии
_____ И. А. Панкратова
Автор работы
студентка группы № 1155
_____ Н. Ю. Бердникова

Томск 2021

Министерство науки и высшего образования Российской Федерации
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)
Институт прикладной математики и компьютерных наук
Кафедра компьютерной безопасности

УТВЕРЖДАЮ
Руководитель ООП
канд. техн. наук, доцент
_____ С. А. Останин
“ 9 ” сентября 2020г.

ЗАДАНИЕ

по подготовке дипломной работы студентке группы 1155 Бердниковой Наталье Юрьевне.

1. Тема дипломной работы: «Условия криптоаналитической обратимости функций».

2. Срок сдачи студентом выполненной дипломной работы:

а) на кафедре: 19 января 2021 года;

б) в ГЭК: 1 февраля 2021 года.

3. Цель работы: исследование криптоаналитически обратимых функций типов $\forall\forall$, $\forall\exists$, $\exists\forall$.

Объекты исследования: криптоаналитически обратимые функции.

Методы исследования: теоретический.

4. Основные разделы работы: поиск тестов обратимости функций, алгоритмы генерации криптоаналитически обратимых функций, алгоритмы построения функций восстановления, подсчет и оценка криптоаналитически обратимых функций.

5. Работа выполняется по заданию кафедры компьютерной безопасности Томского государственного университета.

6. Дата выдачи задания « 01 » сентября 2020г.

Руководитель дипломной работы,
заведующая лабораторией
компьютерной криптографии _____ / И. А. Панкратова/
Задание принял к исполнению,
студентка группы № 1155 _____ / Н. Ю. Бердникова/

РЕФЕРАТ

Дипломная работа содержит 18 страниц, 2 литературных источника.

ОБРАТИМЫЕ ФУНКЦИИ, КРИПТОАНАЛИТИЧЕСКАЯ ОБРАТИМОСТЬ, ТЕСТЫ ОБРАТИМОСТИ ФУНКЦИЙ, ФУНКЦИЯ ВОССТАНОВЛЕНИЯ, ГЕНЕРАЦИЯ ФУНКЦИЙ

Объекты исследования: криптоаналитически обратимые функции.

Цель работы: исследование криптоаналитически обратимых функций типов $\forall\forall$, $\forall\exists$, $\exists\forall$.

Метод исследования: теоретический.

Результат: получены тесты обратимости для типов $\forall\exists$, $\exists\forall$. Разработаны алгоритмы генерации криптоаналитически обратимых функций типа $\forall\forall$, $\forall\exists$, $\exists\forall$ и построения функций восстановления. Доказаны корректность и полнота алгоритмов генерации криптоаналитически обратимых функций типа $\forall\forall$, $\forall\exists$, $\exists\forall$ и построения функций восстановления. Подсчитано и оценено количество криптоаналитически обратимых функций типа $\forall\forall$, $\exists\forall$.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

На сегодняшний день известны некоторые криптосистемы на конечных автоматах и для некоторых из них необходимы обратимые автоматы. В работе [1] были предложены 208 типов обратимости автоматов, многие из которых до сих пор не изучены. Обратимость автомата является частным случаем обратимости функций, исследование которой происходит в рамках данной работы.

1 Цель и задачи

1.1 Цель

Основной целью данной работы является исследование криптоаналитически обратимых функций типов $\forall\forall$, $\forall\exists$, $\exists\forall$.

1.2 Задачи

— Разработать тесты обратимости для типов $\forall\exists$, $\exists\forall$.

— Разработать алгоритмы генерации криптоаналитически обратимых функций типа $\forall\forall$, $\forall\exists$, $\exists\forall$ и построения функций восстановления.

— Доказать корректность и полноту алгоритмов генерации криптоаналитически обратимых функций типа $\forall\forall$, $\forall\exists$, $\exists\forall$ и построения функций восстановления.

— Подсчитать или оценить количество криптоаналитически обратимых функций разных типов.

2 Определения и обозначения

Дано: функция $g(x_1, ..., x_n)$, кванторная последовательность $Q_1x_1...Q_nx_n$ и число $k \in \{1, ..., n\}$, где $Q_k = \forall$. Функция g называется *криптоаналитически обратимой* функцией типа $Q_1x_1Q_2x_2...Q_nx_n$ по переменной x_k, если для нее существует функция восстановления f такая, что формула

$$\mathrm{Q}_1x_1Q_2x_2...Q_nx_n(f(g(x_1, x_2, ..., x_n)) = x_k) \qquad (*)$$

равна истине [2].

Пример 1:

Дано: $g(x_1, x_2), \forall x_1 \exists x_2(f(g(x_1, x_2)) = x_1)$ (**)

Таблица 1 — Иллюстрация к примеру 1

x_1	x_2	$g(x_1, x_2)$	$f(g(x_1, x_2))$
0	0	0	0
0	1	1	1
1	0	1	1
1	1	1	1

Как мы можем видеть, функция f удовлетворяет формуле (**), значит функция g является обратимой функцией по типу $\forall\exists$ по переменной x_1.

Пример 2:

Дано: $g(x_1, x_2), \forall x_1 \exists x_2(f(g(x_1, x_2)) = x_1)$ (**)

Таблица 2 — Иллюстрация к примеру 2

x_1	x_2	$g(x_1, x_2)$	$f_1(g(x_1, x_2))$	$f_2(g(x_1, x_2))$
0	0	0	0	1
0	1	0	0	1
1	0	0	0	1
1	1	0	0	1

Видим, что не существует функций f, удовлетворяющих формуле (**), значит функция g не является обратимой функцией по типу $\forall\exists$ по переменной x_1.

Пример 3:

Дано: $g(x_1, x_2), \exists x_1 \forall x_2(f(g(x_1, x_2)) = x_2)$ (***)

Видим, что функция f удовлетворяет формуле (***), значит функция g является обратимой функцией по типу $\exists\forall$ по переменной x_2.

Таблица 3 — Иллюстрация к примеру 3

x_1	x_2	$g(x_1, x_2)$	$f(g(x_1, x_2))$
0	0	1	0
0	1	0	1
1	0	1	0
1	1	1	0

2.1 Обозначения

D_g - область значения функции $g(x_1, x_2)$ и переменных x_1 и x_2.
$g_a = g|_{x_1=a} = g(a, x_2, ..., x_n)$ - сужение функции g.

3 Криптоаналитическая обратимость функций по двум кванторам

3.1 Криптоаналитически обратимые функции типа ∀∀

Для данного типа обратимости были разработаны алгоритмы, генерирующие криптоаналитически обратимые функции $g(x_1, x_2)$ по переменной x_1 и криптоаналитически обратимые функции $g(x_1, x_2)$ по переменной x_2, с помощью применения леммы 1, являющейся также тестом обратимости для этих функций. Кроме того, были разработаны алгоритмы построения функций восстановления $f(g(x_1, x_2))$. Стоит отметить, что каждый из этих алгоритмов работает только на конечной области определения.

Лемма 1 (Тест обратимости типа ∀∀) [1]. Пусть $k \in \{1, ..., n\}$, $Q_1 = \ldots = Q_n = \forall$, тогда функция восстановления f, удовлетворяющая формуле (*), существует, если и только если

$$\forall x_1 ... \forall x_n \forall y_1 ... \forall y_n (x_k \neq y_k \Rightarrow g(x_1, ..., x_n) \neq g(y_1, ..., y_n)) \quad .$$

3.1.1 Алгоритмы генерации криптоаналитически обратимых функций

Алгоритм 1 (по переменной x_1):

Вход: D_g

Выход: $g(x_1, x_2)$

1. $D'_g := D_g$
2. Для всех $x_1 \in D_g$
   - 2.1. Выбираем случайное m из D'_g
   - 2.2. Для всех $x_2 \in D_g$
     - 2.2.1. $g(x_1, x_2) := m$
   - 2.3. $D'_g := D'_g \setminus \{m\}$
3. Выход: $g(x_1, x_2)$

Алгоритм 2 (по переменной x_2):

Вход: D_g

Выход: $g(x_1, x_2)$

1. $D'_g := D_g$
2. Для всех $x_2 \in D_g$
   - 2.1. Выбираем случайное m из D'_g

2.2. Для всех $x_1 \in D_g$

2.2.1. $g(x_1, x_2) := m$

2.3. $D'_g := D'_g \setminus \{m\}$

3. Выход: $g(x_1, x_2)$

Так как данные алгоритмы идентичны с точностью до перестановки переменных, то доказательства корректности и полноты описаны только для первого алгоритма.

Корректность алгоритма. Второй шаг алгоритма 1 обеспечивает достаточное условие леммы 1 [1] на выходе мы всегда получим криптоаналитически обратимую функцию $\mathrm{g}(x_1, x_2)$ по переменной x_1.

Полнота алгоритма. Пусть дана $g(x_1, x_2)$ - криптоаналитически обратимая функция по типу $\forall\forall$ по переменной x_1. Докажем, что с помощью алгоритма 1 всегда можно построить данную функцию. Во втором шаге рассматриваемого алгоритма происходит заполнение всех сужений функции $g'(x_1, x_2)$ по переменной x_1 различными, случайными значениями из D_g, тогда, если будет выбрано $m = g(x_1, x_2)$, то мы получим функцию $g'(x_1, x_2)$ равную функции $g(x_1, x_2)$.

3.1.2 Подсчет или оценка криптоаналитически обратимых функций

Посчитаем количество функций следующим образом: заметим, что каждое сужение - функция-константа, и значения этих констант для сужений по разным переменным попарно различны, таким образом все функции g можно представить в виде перестановок этих сужений, зная, что количество перестановок равно $n!$, где $n = |D_g|$, получаем, что количество всех g равно $n!$.

3.1.3 Алгоритмы построения функции восстановления

Алгоритм 3 (для обратимых функций $\mathbf{g(x_1, x_2)}$ по типу $\forall\forall$ по переменной $\mathbf{x_1}$):
Вход: $g(x_1, x_2)$
Выход: $f(g(x_1, x_2))$

1. Для всех $x_1 \in D_g$

1.1. x_2 - любое случайное $\in D_g$

1.2. $f(g(x_1, x_2)) := x_1$

2. Выход: $f(g(x_1, x_2))$

Алгоритм 4 (для обратимых функций $\mathbf{g(x_1, x_2)}$ по типу $\forall\forall$ по переменной $\mathbf{x_2}$):

Вход: $g(x_1, x_2)$
Выход: $f(g(x_1, x_2))$

1. Для всех $x_2 \in D_g$

   1.1. x_1 - любое случайное $\in D_g$

   1.2. $f(g(x_1, x_2)) := x_2$

2. Выход: $f(g(x_1, x_2))$

Так как данные алгоритмы полностью идентичны, за исключением только того, по какой переменной функция $g(x_1, x_2)$ является обратимой, то будем рассматривать корректность только для алгоритма 3.

Корректность алгоритма. Пусть дана криптоаналитически обратимая функции $g(x_1, x_2)$ по типу $\forall\forall$ по переменной x_1. Докажем, что построенная на выходе функция f - функция восстановления. Положим, что это не так, тогда на выходе мы получим функцию f такую, что $f(a) \neq f(b)$ при $a = b$, но это невозможно, так как это противоречит шагу 1.2 и условию леммы 1.

3.2 Криптоаналитически обратимые функции типа $\forall\exists$

3.2.1 Тест обратимости

В качестве теста обратимости функций было выведено и доказано следующее утверждение:

Утверждение 1 (Тест обратимости типа $\forall\exists$). Пусть $D_g = \{x_{1_1}, ..., x_{1_n}\}$. Тогда $g(x_1, x_2)$ обратима типа $\forall\exists$, если и только если существуют $x_{2_1}, ..., x_{2_n}$, такие, что
$|\{g(x_{1_1}, x_{2_1}), ..., g(x_{1_n}, x_{2_n})\}| = n$.

Доказательство:

$\Rightarrow$
$g(x_1, x_2)$ - обратима $\Rightarrow$
$\exists f(\forall x_1 \exists x_2(f(g(x_1, x_2)) = x_1)) \Rightarrow$
$\exists f(\exists x_{2_1}(f(g(x_{1_1}, x_{2_1})) = x_{1_1}), ..., \exists x_{2_n}(f(g(x_{1_n}, x_{2_n})) = x_{1_n}) \Rightarrow$
$\exists f(\exists x_{2_1}, ..., x_{2_n}|\{f(g(x_{1_1}, x_{2_1})), ..., f(g(x_{1_n}, x_{2_n}))\}| = n) \Rightarrow$
$\exists x_{2_1}, ..., x_{2_n}(|\{g(x_{1_1}, x_{2_1}), ..., g(x_{1_n}, x_{2_n})\}| = n)$

$\Leftarrow$ Пусть $g(x_1, x_2)$ - функция, удовлетворяющая условиям достаточности утверждения 1, тогда по определению функции восстановления f, она будет получена с помощью следующей формулы $f(g(x_{1_i}, x_{2_i})) = x_{1_i}, i = 1, ..., n$.

3.2.2 Алгоритм построения функции восстановления

В качестве алгоритма построения функции восстановления f для данного типа обратимости используется обход дерева в глубину.

Обход дерева происходит следующим образом: рекурсивно перебираются все ребра, исходящие из рассматриваемой вершины. Если ребро ведет в вершину, которая не была рассмотрена ранее, то перебираем ребра от этой нерассмотренной вершины, а после возвращаемся и продолжаем перебирать рёбра. Возвращаемся в случае, когда мы дошли до листа дерева и не получили на выходе необходимую нам функцию восстановления f.

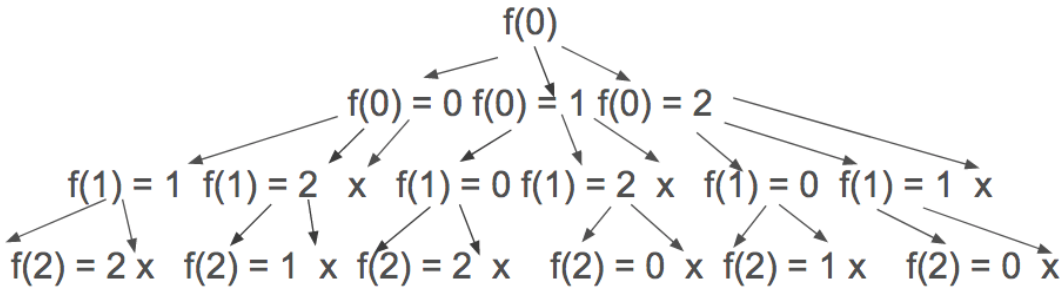


Рисунок 1 — Схема дерева построения функции f

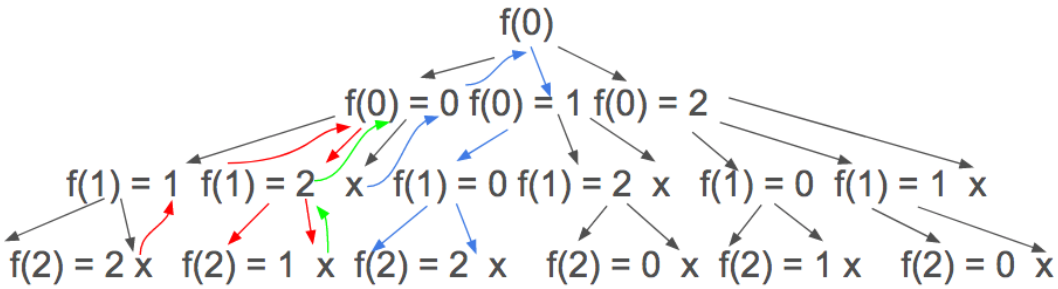


Рисунок 2 — Обход дерева построения функции f

Рассмотрим пример (таблица 4) построения функции восстановления f для функции $g(x_1, x_2)$, обратимой по типу $\forall\exists$:

Шаг 1. Заходим в первую нерассмотренную вершину $f(0) := 0$.

Шаг 2. Заходим в следующую нерассмотренную вершину $f(1) := 1$.

Шаг 3. Заходим в следующую нерассмотренную вершину $f(2) := 2$, которая является листом дерева, так как, на выходе мы не получили функцию восстановления f, и для вершины $f(1) := 1$ больше нет ребер, то идем в следующую ветку от вершины $f(0) := 0$ и продолжаем поиск.

Шаг 4. Заходим в ранее нерассмотренную вершину $f(1) := 2$, так как для нее еще есть ребра, то продолжаем поиск.

Шаг 5. Заходим в ранее нерассмотренную вершину $f(2) := 1$, которая является листом и функция восстановления f найдена.

Таблица 4 — Пример построения функции восстановления $f(g(x_1, x_2))$

x_1	x_2	$g(x_1, x_2)$	$f_1(g(x_1, x_2))$	$f_2(g(x_1, x_2))$	$f_3(g(x_1, x_2))$
0	0	0	0	0	0
0	1	1	-	1	2
0	2	0	0	0	0
1	0	2	-	-	1
1	1	0	0	0	0
1	2	0	0	0	0
2	0	1	-	1	2
2	1	1	-	1	2
2	2	0	0	0	0

3.2.3 Алгоритм генерации криптоаналитически обратимой функции по типу $\forall\exists$ по переменной x_1

Применим доказанное выше утверждение для построения алгоритма генерации обратимых функций по типу $\exists\forall$ по переменной x_1, суть которого в следующем.

Для всех сужений функции g по переменной x_1 случайно выбирается значение x_2, затем каждому из этих сужений с выбранными для них x_2 присваиваются случайные значения без повторений из D_g, а всем оставшимся $g(x_1, x_2)$ присваиваются случайные значения из D_g.

Алгоритм 5

Вход: D_g

Выход: $g(x_1, x_2)$

1. $D'_g := D_g$
2. Для каждого $x_1 \in D_g$
   - 2.1. $x_2 := rand(D_g)$
   - 2.2. $g(x_1, x_2) := rand(D'_g)$
   - 2.3. $D'_g := D'_g \setminus \{g(x_1, x_2)\}$
   - 2.4. Для каждого $m \neq x_2$
     - 2.4.1. $g(x_1, m) := rand(D_g)$
3. Выход: $g(x_1, x_2)$

Корректность алгоритма. Второй шаг алгоритма обеспечивает достаточное условие утверждения 1 на выходе мы всегда получим криптоаналитически обратимую функцию $g(x_1, x_2)$ по переменной x_1.

Полнота алгоритма. Пусть дана $g(x_1, x_2)$ - криптоаналитически обратимая функция по типу $\forall\exists$ по переменной x_1. Докажем, что с помощью данного алгоритма, всегда можно сгенерировать данную функцию. С помощью шагов 2.1-2.3 происходит заполнение случайного $g'_{x_1}(x_2)$ для каждого x_1 из D_g различными, случайными значениями из D_g, оставшиеся же значения функции $g'(x_1, x_2)$ заполняются случайными значениями из D_g на шаге 2.4.1. Таким образом, на некотором шаге генерации криптоаналитически обратимых функций по типу $\forall\exists$ по переменной x_1 мы получаем функцию $g'(x_1, x_2)$, равную данной функции $g(x_1, x_2)$.

3.3 Криптоаналитически обратимые функции типа $\exists\forall$

3.3.1 Тест обратимости

Утверждение 2 (Тест обратимости типа $\exists\forall$). Функция $g(x_1, x_2)$ - криптоаналитически обратима по типу $\exists\forall$ по переменной x_2 тогда и только тогда, когда $\exists a|D_{g_a}| = |D_g|$.

Доказательство:

$\Rightarrow$

$g(x_1, x_2)$ - обратима $\Rightarrow$
$\exists f(\exists x_1 \forall x_2(f(g(x_1, x_2)) = x_2)) \Rightarrow$
$\exists f \exists x_1(\forall x_2(f(g_{x_1}(x_2)) = x_2)) \Rightarrow$
Т.к $\forall x_2(f(g_{x_1}(x_2))$ принимают различные значения из $D_g \Rightarrow$
$\forall x_2(g_{x_1}(x_2)$ также принимают различные значения из $D_g \Rightarrow$
$\exists x_1(|D_{g_{x_1}}| = |D_g|)$

$\Leftarrow$

Докажем, что для любой функции $g(x_1, x_2)$, удовлетворяющая условию достаточности, мы всегда можем построить функцию восстановления $f(g(x_1, x_2))$ с помощью алгоритма, суть которого в следующем.

Дано: функция $g(x_1, x_2)$, удовлетворяющая утверждению.

Найти: функцию восстановления $f(g(x_1, x_2))$.

Находим такое сужение функции $g(x_1, x_2)$ по переменной x_1, что мощность области значения этого сужения равна мощности области значения функции $g(x_1, x_2)$, далее для каждого x_2 из сужения функции $g(x_1, x_2)$ присваиваем функции $f(g_{x_1}(x_2))$ значение x_2 и запоминаем, чему равна функция f от $g_{x_1}(x_2)$. Затем, для всех остальных сужений функции $g(x_1, x_2)$ по переменной x_1 присваиваем функции f от $g_{x_1}(x_2)$ значения, которые мы запомнили ранее. Более подробно все действия описаны в алгоритме, приведенном ниже.

3.3.1.1 Алгоритм построения функции восстановления

Алгоритм 6. Алгоритм построения функции восстановления $f(g(x_1, x_2))$ для криптоаналитически обратимой функции $g(x_1, x_2)$ по типу $\forall\exists$.
Вход: функция $g(x_1, x_2)$, удовлетворяющая утверждению
Выход: функция восстановления $f(g(x_1, x_2))$

1. m - вектор длины $|D_g|$
2. Для каждого $x_1 \in D_g$
   - 2.1. Если $|D_{g_{x_1}}| = |D_g|$
     - 2.1.1. Для каждого $x_2 \in D_g$
       - 2.1.1.1. $f(g(x_1, x_2)) := x_2$
       - 2.1.1.2. $m[g(x_1, x_2)] := x_2$
     - 2.1.2. Для каждого $k \in D_g \setminus \{x_1\}$
       - 2.1.2.1. Для каждого $x_2 \in D_g$
       - 2.1.2.2. $f(g(k, x_2)) := f(m[g(k, x_2)])$
     - 2.1.3. Выход $f(g(x_1, x_2))$
3. Иначе продолжаем поиск

Корректность алгоритма. Пусть дана $g(x_1, x_2)$, удовлетворяющая достаточному условию утверждения 2, предположим, что для нее нельзя построить функцию восстановления $f(g(x_1, x_2))$. Согласно достаточному условию утверждения 2 $\exists a|D_{g_a}| = |D_g|$. Пусть $\exists b, c \in g_a : f(c_l) = f(c_m)$, тогда $c_l = c_m \Rightarrow g(x_1, x_2)$ не удовлетворяет достаточному условию утверждения 2. ■

3.3.2 Алгоритм генерации криптоаналитически обратимой функции по типу $\exists\forall$ по переменной x_2

Применим доказанное выше утверждение для построения алгоритма генерации обратимых функций по типу $\exists\forall$ по переменной x_2, суть которого заключается в следующем.

Выбираем случайное значение x_1 из D_g, затем для каждого $x_2 \in D_g$ присваиваем сужению $g_{x_1}(x_2)$ функции $g(x_1, x_2)$ значение из D_g в случайном порядке, так чтобы $|D_{g_{x_1}}| = |D_g|$. Далее, оставшиеся сужения заполняем случайными значениями из D_g.

Алгоритм 7
Вход: D_g
Выход: $g(x_1, x_2)$

1. $D'_g := D_g$

2. Выбираем случайное x_1 из D_g

3. Для каждого $x_2 \in D_g$

   3.1. $g(x_1, x_2) := rand(D'_g)$

   3.2. $D'_g := D'_g \setminus \{g(x_1, x_2)\}$

4. Для каждого $m \in D_g \setminus \{x_1\}$

   4.1. Для каждого $x_2 \in D_g$

      4.1.1. $g(m, x_2) := rand(D_g)$

5. Выход: $g(x_1, x_2)$

Корректность алгоритма. В шаге 3 выполняется условие утверждения 2, значит, с помощью данного алгоритма мы всегда можем построить криптоаналитически обратимую функцию по типу $\exists\forall$ по переменной x_2.

Полнота алгоритма. Пусть $g(x_1, x_2)$ криптоаналитически обратимая функция по типу $\exists\forall$ по переменной x_2. Докажем, что с помощью описанного выше алгоритма мы всегда сможем построить данную функцию. Во втором шаге алгоритма мы выбираем случайное x_1 из D_g, так как x_1 выбирается случайно, то на некотором шаге генерации функций с помощью данного алгоритма мы получим необходимое нам x_1, такое, что $\forall x_2 f(g(x_1, x_2)) = x_2$. На 3 шаге алгоритма $g'_{x_1}(x_2)$ заполняется различными случайными значениями из D_g, т.е на некотором шаге генерации функций значения $g'_{x_1}(x_2)$ совпадут со значениями $g_{x_1}(x_2)$. Оставшиеся значения функции $g'(x_1, x_2)$ заполняются случайными значениями из D_g, следовательно, на некотором шаге генерации криптоаналитически обратимых функций по типу $\exists\forall$ по переменной x_2 мы построим функцию $g'(x_1, x_2)$, равную функции $g(x_1, x_2)$.

3.3.3 Подсчет или оценка криптоаналитически обратимых функций

Для оценки количества функций данного типа разобьем все эти функции на блоки по следующему принципу: каждый блок содержит функции с сужениями по зафиксированной переменной x_1, мощность области значений которых равна мощности D_g, тогда всего таких блоков будет $|D_g|$.

Для каждого блока все его функции, в которых значения зафиксированных сужений равны можно распределить по подмножествам, всего таких подмножеств будет $|D_g|!$, а в каждом подмножестве будет $|D_g|^{(|D_g|-1)*|D_g|}$ функций.

Тогда получаем следующую оценку: количество криптоаналитически обратимых функций по переменной $x_2 \leqslant |D_g|! * |D_g|^{|D_g|^2-|D_g|+1}$.

ЗАКЛЮЧЕНИЕ

В ходе выполнения данной дипломной работы были получены тесты обратимости для типов $\forall\exists$, $\exists\forall$, разработаны алгоритмы генерации криптоаналитически обратимых функций типа $\forall\forall$, $\forall\exists$, $\exists\forall$ и построения функций восстановления, доказаны корректность и полнота алгоритмов генерации криптоаналитически обратимых функций типа $\forall\forall$, $\forall\exists$, $\exists\forall$ и построения функций восстановления, подсчитано количество криптоаналитически обратимых функций для типов $\forall\forall$ и оценено количество криптоаналитически обратимых функций для типов $\exists\forall$.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. *Agibalov G. P.* Cryptanalytic concept of finite automaton invertibility with finite delay // Прикладная дискретная математика. №44. — Томск, 2019. — с. 34—42.
2. *Agibalov G. P.* Cryptanalytical finite automaton invertibility with finite delay // Прикладная дискретная математика. №46. — Томск, 2019. — с. 27—37.

АНТИПЛАГИАТ
ТВОРИТЕ СОБСТВЕННЫМ УМОМ

Отчет о проверке на заимствования №1

Автор: Бердникова Наталья nat.berdnikova97@gmail.com / ID: 8712361
Проверяющий: Бердникова Наталья (nat.berdnikova97@gmail.com / ID: 8712361)

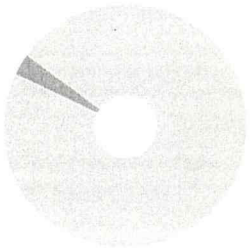
Отчет предоставлен сервисом «Антиплагиат»- http://users.antiplagiat.ru

ИНФОРМАЦИЯ О ДОКУМЕНТЕ

№ документа: 1
Начало загрузки: 23.01.2021 16:49:51
Длительность загрузки: 00:00:00
Имя исходного файла: Diploma.pdf
Название документа: Diploma
Размер текста: 19 кБ
Символов в тексте: 19258
Слов в тексте: 2586
Число предложений: 88

ИНФОРМАЦИЯ ОБ ОТЧЕТЕ

Последний готовый отчет (ред.)
Начало проверки: 23.01.2021 16:49:52
Длительность проверки: 00:00:02
Комментарии: не указано
Модули поиска: Модуль поиска Интернет



ЗАИМСТВОВАНИЯ	САМОЦИТИРОВАНИЯ	ЦИТИРОВАНИЯ	ОРИГИНАЛЬНОСТЬ
3,15%	0%	0%	96,85%

Заимствования — доля всех найденных текстовых пересечений, за исключением тех, которые система отнесла к цитированиям, по отношению к общему объему документа.
Самоцитирования — доля фрагментов текста проверяемого документа, совпадающий или почти совпадающий с фрагментом текста источника, автором или соавтором которого является автор проверяемого документа, по отношению к общему объему документа.
Цитирования — доля текстовых пересечений, которые не являются авторскими, но система посчитала их использование корректным, по отношению к общему объему документа. Сюда относятся оформленные по ГОСТу цитаты; общеупотребительные выражения; фрагменты текста, найденные в источниках из коллекций нормативно-правовой документации.
Текстовое пересечение — фрагмент текста проверяемого документа, совпадающий или почти совпадающий с фрагментом текста источника.
Источник — документ, проиндексированный в системе и содержащийся в модуле поиска, по которому проводится проверка.
Оригинальность — доля фрагментов текста проверяемого документа, не обнаруженных ни в одном источнике, по которым шла проверка, по отношению к общему объему документа.
Заимствования, самоцитирования, цитирования и оригинальность являются отдельными показателями и в сумме дают 100%, что соответствует всему тексту проверяемого документа.
Обращаем Ваше внимание, что система находит текстовые пересечения проверяемого документа с проиндексированными в системе текстовыми источниками. При этом система является вспомогательным инструментом, определение корректности и правомерности заимствований или цитирований, а также авторства текстовых фрагментов проверяемого документа остается в компетенции проверяющего.

№	Доля в отчете	Источник	Ссылка	Актуален на	Модуль поиска
[01]	0,82%	Шаблоны титульных листов ВКР	http://tsu.ru	10 Дек 2018	Модуль поиска Интернет
[02]	0%	http://www.lib.tsu.ru/win/produkzija/metodichka/pril_A_4.pdf	http://lib.tsu.ru	10 Июн 2019	Модуль поиска Интернет
[03]	0,78%	http://vital.lib.tsu.ru/vital/access/services/Download/vital:10545/SOURCE01	http://vital.lib.tsu.ru	24 Янв 2020	Модуль поиска Интернет

Еще источников: 11
Еще заимствований: 1,56%