Приложение 2. Перечень тем исследований

К участию в Конкурсе принимаются работы на русском или английском языке, посвященные решению хотя бы одной из следующих задач для хотя бы одной из следующих функций $f(IV, M)$ со значениями в V_n.

Функции

1. Функции хэширования, определяемые ГОСТ Р 34.11-2012. В этом случае $n = 256$ или $n = 512$, значения $IV \in V_{512}$ фиксированы и определены в п. 5.1 стандарта, если не оговорено противное, и $M \in V^*$.
2. Функции хэширования, определяемые ГОСТ Р 34.11-2012, без завершающих или одного из завершающих преобразований (шаги 3.3.-3.6). В этом случае, так же, как в предыдущем, $n = 256$ или $n = 512$, значения $IV \in V_{512}$ фиксированы и определены в п. 5.1 стандарта, если не оговорено противное, и $M \in V^*$.
3. Функции сжатия функций хэширования, определяемых ГОСТ Р 34.11-2012. В этом случае $n = 512$, значение $IV \in V_{512}$ произвольно и фиксировано, если не оговорено противное, и $M \in V_{512}$.
4. Усечённые (round-reduced) варианты указанных выше функций, т.е. соответствующие использованию в преобразовании $E(K, m)$ меньшего числа преобразований $LPSX$, чем определено в ГОСТ Р 34.11-2012.

Задачи

1. Обращение (построение прообраза, preimage attack). По заданному значению $h \in V_n$ найти значение *M*, такое, что $f(IV, M) = h$.
2. Построение коллизии. Найти два различных значения *M* и M', таких, что $f(IV, M) = f(IV, M')$.
3. Построение второго прообраза (second preimage attack). По заданному значению *M* найти отличное от *M* значение M', такое, что $f(IV, M) = f(IV, M')$.
4. Построение псевдо-прообраза (pseudo-preimage attack). По заданному значению $h \in V_n$ найти значения *IV* и *M*, такие, что $f(IV, M) = h$.
5. Построение условно-свободной коллизии (collision attack for different IV, semi-free-start collision attack). Найти значение *IV* и два различных значения M, M', такие, что $f(IV, M) = f(IV, M')$.
6. Построение псевдо-коллизии (pseudo-collision attack, free-start collision attack). Найти два значения IV, IV' и два различных значения M, M' такие, что $f(IV, M) = f(IV', M')$.
7. Построение второго псевдо-прообраза (second pseudo-preimage attack, free-start target attack). По заданному значению *M* найти значение IV' и отличное от *M* значение M', такие, что $f(IV, M) = f(IV', M')$.
8. Построение мульти-коллизии (построение *r*-коллизии). Найти попарно различные значения $M_1, \ldots, M_r$, такие, что $f(IV, M_1) = \ldots = f(IV, M_r)$

9. Построение мульти-прообраза (построение *r*-прообраза). По заданному значению $h \in V_n$ найти попарно различные значения $M_1,\ldots,M_r$, такие, что $f(IV,M_1)=\ldots=f(IV,M_r)=h$.
10. Построение второго мульти-прообраза (построение второго *r*-прообраза). По заданному значению *M* найти отличные от *M* и попарно различные значения $M_1,\ldots,M_r$, такие, что $f(IV,M_1)=\ldots=f(IV,M_r)=f(IV,M)$.
11. Построение почти-прообраза. По заданному значению $h \in V_n$ найти значение *M*, такое, что сумма $f(IV,M)\oplus h$ имеет небольшой вес Хэмминга.
12. Построение почти-коллизии (near-collision attack). Найти два различных значения *M* и M', таких, что сумма $f(IV,M)\oplus f(IV,M')$ имеет небольшой вес Хэмминга.
13. Построение второго почти-прообраза. По заданному значению *M* найти отличное от *M* значение M', такое, что сумма $f(IV,M)\oplus f(IV,M')$ имеет небольшой вес Хэмминга.
14. Расширение сообщения (length-extension attack, только для функций хэширования и их усеченных вариантов). По заданным значениям $|M|$, $f(IV,M)$ найти некоторое значение M', для которого вычислить $f(IV,M\|M')$.
15. Построение прообраза при заданном префиксе сообщения и заранее выбранном значении функции (chosen target force prefix, CTFP, Nostradamus attack, только для функций хэширования и их усеченных вариантов). Задача состоит из двух этапов. На первом этапе требуется построить и предъявить некоторое значение $h \in V_n$. На втором этапе, по заданному значению *M*, выбираемому из некоторого заранее известного множества значений, требуется найти такое значение M', что $f(IV,M\|M')=h$.
16. Построение алгоритма различения (distinguishing attack). Построить алгоритм, позволяющий отличить функцию $f(IV,M)$ от случайно и равновероятно выбранной функции.