А.А. Петросян, М.Е. Бурлаков, В.В. Бондаренко

АНАЛИЗ СПОСОБА ОПТИМИЗАЦИИ ДИФФЕРЕНЦИАЛЬНОГО КРИПТОАНАЛИЗА

(Самарский национальныйисследовательский университет
имени академика С.П. Королёва)

Введение. В настоящее время существует большое количество криптографических алгоритмов (КА), обеспечивающих защиту всевозможных информационных систем. Существует множество техник, позволяющих проанализировать устойчивость КА к криптоанализу. К таковым можно отнести: дифференциальный анализ, линейный анализ и т.д. [1,2]. Зачастую, процесс криптоанализа требует больших затрат вычислительных ресурсов, что приводит к усложнению проверки стойкости КА к криптоанализу.

В данной статье будет рассмотрен и проанализирован один из способов оптимизации дифференциального криптоанализа, предложенный в статье «*A Genetic Algorithm for Cryptanalysis of DES*-8» [3].

Дифференциальный криптоанализ представляет собой атаку на основе подобранного открытого текста. Этот тип атаки основан на *n*-раундовых характеристиках. Характеристика имеет определенную разницу открытых текстов ΩP, определенную разницу на *n*-том раунде шифрования ΩT и вероятность P того, что пара открытых текстов в результате шифрования имеет различия, совпадающие с указанными в характеристике.

Пара открытых текстов с *XOR*-разницей ΩP и с *XOR*-разницей шифртекстов ΩT, соответствующей характеристике, называется **правильной парой** [1]. С помощью правильной пары можно вычислить ключи-кандидаты. Ключ-кандидат, который в результате вычислений появляется чаще всего, считается исходным ключом [1].

Таким образом, главной целью криптоаналитика является получение правильных пар. Но с ростом количества раундов шифрования уменьшается вероятность появления правильной пары. Поэтому одним из способов оптимизации ДК может является увеличение вероятности появления правильной пары.

Генетический алгоритм (ГА) - это эвристический алгоритм поиска, используемый для решения задач оптимизации и моделирования путём случайного подбора, комбинирования и вариации искомых решений [4]. **Хромосома** в ГА - вектор значений, определяющих некоторое решение поставленной задачи [4]. **Приспособленность** - значение, определяющее, насколько хорошо закодированное в хромосоме решение справляется с поставленной задачей [4]. **Популяция** - конечное множество хромосом. Изначально генерируется первая популяция. Каждая хромосома популяции оценивается с помощью функции приспособленности. При помощи функций скрещивания и мутации образуется новая популяция решений. Для хромосом новой популяции также вычисляется значе-

ние приспособленности, и затем производится отбор лучших решений в следующее поколение [4].

Описание исследуемой статьи. В статье предлагается вычислять правильные пары с помощью генетических алгоритмов, где хромосомами будут являться открытые тексты.

В качестве функции приспособленности предлагается использовать следующую функцию [3]:

$$Fitness(\Omega_T, T') = 1 - \frac{H_d(\Omega_T, T')}{n},$$

где H_d – расстояние по Хеммингу, n – длина хромосомы (64 бит).

Алгоритм генерации правильных пар следующий:

1. Сгенерировать начальную популяцию, где каждая хромосома представляет с собой открытый текст P;
2. Для каждой хромосомы:
   - вычислить $P^* = \Omega P \oplus P$;
   - вычислить $T' = T \oplus T^* = C(P) \oplus C(P^*)$;
   - вычислить значение функции приспособленности;
   - если значение функции приспособленности больше 0.5, извлечь вероятные биты ключа для данной пары.
3. Применить функцию скрещивания;
4. Применить функцию мутации;
5. Сгенерировать следующую популяцию;
6. Повторить все действия, начиная с шага 2, пока некоторые значения битов ключа не будут предложены значительно больше других.

Недостатки подхода

1. Маленькая область значений функции приспособленности:

Утверждение: Мощность области значений представленной функции приспособленности равна 65.

Доказательство: Так как n и ΩT во время вычислений не меняются, то единственной переменной в функции приспособленности является T'. T' в свою очередь является одним из аргументов дистанции по Хэммингу. Дистанция по Хэммингу определяет, на скольких позициях двух слов одинаковой длины отличаются значения. Длина T' и ΩT равна 64. Соответственно, T' и ΩT могут отличаться от 0 до 64 битами, поэтому $H_d(\Omega T, T')$ имеет 65 значений, следовательно, функция приспособленности имеет 65 значений.

В то время как мощность области определения равна 2^{32}. Из этого следует что в среднем у $2^{27}/65$ хромосом будут одинаковые значения приспособленности, что не дает хорошей сходимости.

2. Рельеф функции приспособленности не является гладким / функция приспособленности имеет огромное количество точек локального экстремума:

Утверждение: Незначительное изменение хромосомы приведет к резкому изменению значения функции приспособленности.

Доказательство: *DES* обладает **строгим критерием лавинного эффекта**, что обозначает, что изменение любого бита открытого текста приведет к изменению любого выходного бита с вероятностью ½ [5].

Пусть P_1 – хромосома, тогда $P_1^* = \Omega P \oplus P_1$. P_2 – хромосома, полученная из P_1 изменением одного бита, тогда $P_2^* = \Omega P \oplus P_2$.

Таким образом P_1 и P_2, P_1^* и P_2^* попарно отличаются одним битом.

Пусть $T_1 = C(P_1), T_2 = C(P_2), T_1^* = C(P_1^*), T_2^* = C(P_2^*)$.

Пусть $T_1' = T_1 \oplus T_1^*, T_2' = T_2 \oplus T_2^*$.

P_1 и P_2 отличаются одним битом, то по строгому критерию лавинного эффекта, T_1 и T_2 значительно отличаются. P_1^* и P_2^*. отличаются одним битом, то по строгому критерию лавинного эффекта, T_1^* и T_2^* значительно отличаются. Откуда следует, что T_1' значительно отличается от T_2'.

В пункте 5.2. было выяснено, что значение функции приспособленности зависит только от T'. Откуда следует, что значения функции приспособленности для P_1 и P_2 будут значительно отличаться.

Таким образом изменение одного бита хромосомы приведет к резкому изменению значения функции приспособленности.

3. Алгоритм требует неоднократные запросы к шифрующей системе

Так как для вычисления функции приспособленности требуется зашифровать пару открытых текстов, то криптоаналитку потребуется иметь непрерывный доступ к шифрующей системе.

4. Авторы не описывают по каким алгоритмам производится скрещивание и мутация хромосом, которые необходимы для работы генетических алгоритмов.

Реализация. Описанный алгоритм оптимизации был реализован на языке *Java* с использованием библиотеки *Jenetics* [6]. Так как авторы не описывают функции скрещивания и мутации, в качестве функции скрещивания было взято точечное скрещивание. В качестве функции мутации было взято случайное изменение одного бита открытого текста.

Атака производилась на шифр *DES*, укороченный до 8 раундов. Для атаки использовалась двухраундовая характеристика с $\Omega P = 1960000000000000$, $\Omega T = 1960000000000000$ и $P = \frac{1}{243}$. Размер популяции = 5, вероятность скрещивания = 0.6, вероятность мутации = 0.2.

На рисунке 1 изображен график изменения функции приспособленности. По оси абсцисс отмечен номер популяции, по оси ординат – лучшее значение функции приспособленности в популяции. Как видно на графике, приспособленность растет очень медленно и не равномерно, что говорит о том, что приспособленность растет только благодаря случайности мутации. Сходимости к оптимальному решению не наблюдается. Таким образом, реализация алгоритма не дала видимого прироста вероятности появления правильной парой.

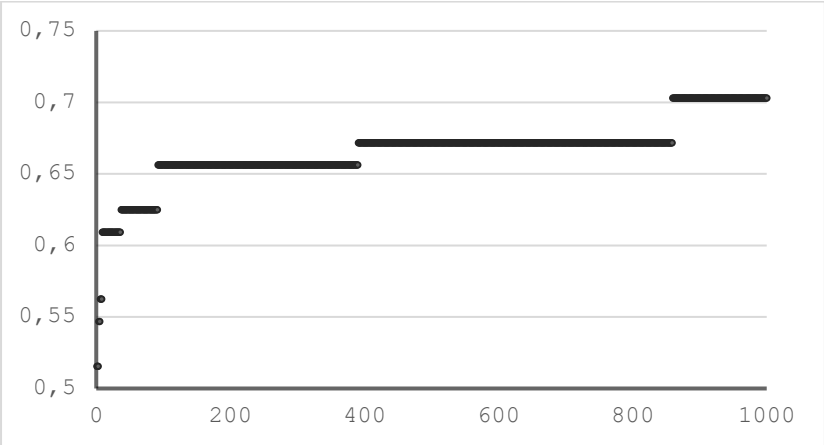

Рис. 1. График изменения приспособленности в зависимости от поколения

Вывод. В данное работе был проанализирован один из способов оптимизации дифференциального криптоанализа генетическим алгоритмом. В процессе анализа были выявлены недостатки описанного подхода, которые в рамках данной работы были теоретически обоснованы. Для дополнительного анализа, алгоритм был реализован и протестирован на практике. Заметного прироста частоты появления правильных пар не наблюдается.

Учитывая описанные доводы, можно утверждать, что способ оптимизации дифференциального анализа, предложенный в статье «*A Genetic Algorithm for Cryptanalysis of DES*-8» [1], не является рабочим для реализации атаки и требует доработки.

Литература

1. Biham E. Differential Cryptanalysis of DES-like Cryptosystems / Eli Biham, Adi Shamir // The Weizmann Institute of Science Department of Applied Mathematics, 1990. − P. 8-38.

2. Matsui M. Linear Cryptanalysis Method for DES Cipher // Advances in Cryptology-EUROCRYPT'93, 1994. − P. 386-397.

3. Hasan Mohammed Hasan Husein A Genetic Algorithm for Cryptanalysis of DES-8 / Hasan Mohammed Hasan Husein, Bayoumi I. Bayoumi, Fathy Saad Holail, Bahaa Eldin M. Hasan, Mohammed Z. Abd El-Mageed // International Journal of Network Security, Vol.5, No.2, PP.213–219, Sept. 2007

4. Генетический алгоритм [Электронный ресурс], − Режим доступа: https://ru.wikipedia.org/.

5. Бабенко Л. К. Современные алгоритмы блочного шифрования и методы их анализа: учеб. пособие для студентов вузов, обучающихся по группе специальностей в обл. информ. Безопасности / Л. К. Бабенко, Е. А. Ищукова. — М.: Гелиос АРВ, 2006. — 376 с.

6. Jenetics: Java Genetic Algorithm Library [Электронный ресурс], − Режим доступа: http://jenetics.io/.