УДК 681.3

Фаль О.М.

АТАКИ НА ПОТОКОВІ ШИФРИ, ЩО ПОЄДНУЮТЬ СТАТИСТИЧНІ ТА АЛГЕБРАЇЧНІ МЕТОДИ

Статистичне тестування на основі алгебраїчної нормальної форми булевої функції

Алгебраїчною нормальною формою (АНФ) булевої функції f будемо називати многочлен від багатьох змінних, що задається виразом:

$$f(x_1, x_2, \ldots, x_n) = \oplus_{u \in F_2^n} a_u x^u, a_u \in F_2,$$

в якому $u=u(u_1, u_2, \ldots, u_n)$, $x^u = \prod_{i=1}^{n} x_i^{u_i}$

Коефіцієнти a_u обчислюються, виходячи з таблиці істинності функції f наступним чином:

$$a_u = \bigoplus_{x \le u} f(x),$$

де значок $\le$ позначає частковий порядок на булевій решітці, тобто $\alpha \le \beta$ тоді і тільки тоді, коли $\alpha_s \le \beta_s$ для усіх $1 \le s \le n$. Кажуть, що одночлен $a_u x^u$ АНФ має степінь k, якщо $a_u=1$ та $wt(u)=k$, де $wt(\cdot)$ - вага Хемінга.

В роботі [1] запропоновано новий спосіб оцінювання статистичних характеристик криптографічних систем, які впливають на їх стійкість. Автор вперше запропонував використати в ньому АНФ булевої функції, що описує роботу шифра З визначення випадкової булевої функції випливає, що її АНФ має в середньому 2^{n-1} одночленів. Для кожного k, $0 \le k \le n$, в АНФ міститься в середньому $\frac{1}{2}\binom{n}{k}$ одночленів степеня k.

Використовуючи центральну граничну теорему (ЦГТ) для бернулівського розподілу ймовірностей, можна стверджувати, що розподіл випадкової величини n_k кількості одночленів порядку k в АНФ випадкової булевої функції апроксимується нормальним розподілом з параметрами $En_k = \frac{1}{2}\binom{n}{k}$, $Dn_k = \frac{1}{4}\binom{n}{k}$.

Розглянемо деяку симетричну криптосистему. Нехай $K=(k_0, k_1, \ldots, k_{n-1})$ позначає її симетричний секретний ключ. Потоковий шифр можна уявляти собі наступним чином: кожний біт i на виході, що генерується, виходячи з секретного ключа K, є однозначно визначеною АНФ. Іншими словами, N-бітова послідовність на виході може бути описана сімейством N булевих функцій $(f_t(k)_{0 \le t \le N-1} = (f_0(k), f_1(k), \ldots, f_{N-1}(k))$, де $f_i(k)$ позначає i-й біт, що виробляється системою і моделюється як многочлен від змінних k_i (АНФ). Кожний біт на виході є значенням, булевої функції $f_t: F_2^n \to F_2$. Далі ми будемо вважати, що біти (значення булевих функцій) є статистично незалежними. Це припущення є загально прийнятим у криптоаналізі шифрів.

Повну АНФ неможливо обчислити при великих n, так як вона містить в середньому 2^{n-1} одночленів. Для подальших тестів ми сфокусуємо нашу увагу на одночленах степенів не вищих за 3. Тобто, ми будемо обчислювати часткову АНФ, що складається з одночленів, степінь яких не перевищує 3. Будемо позначати через n_d кількість одночленів, степінь яких в точності дорівнює d, через H_0^d статистичну гіпотезу, що число n_d має нормальний розподіл з параметрами $\frac{1}{2}\binom{n}{d}$, $\frac{1}{4}\binom{n}{d}$.

Розглянемо спочатку тест афінної константи. Тоді H_0^0 позначає відповідну нульову гіпотезу. Згідно з викладеним вище, ймовірність того, що афінна константа a_0 АНФ рівна 1, дорівнює $\frac{1}{2}$. Це означає, що число булевих функцій, у яких a_0=1 в їхніх АНФ апроксимується розподілом $N\left(\frac{N}{2},\frac{\sqrt{N}}{2}\right)$, де N є числом булевих функцій, що описують послідовність, а отже їх АНФ.

Нехай X_s, кількість випадків a_0=1, позначає статистику у виборці S, елементами якої є послідовності N АНФ. Тест афінної константи задається наступним чином:

1. З S обчислюємо X_s.

2. Зафіксуємо помилку першого роду через α (тобто ймовірність відхилення H_0^0 за умови, що вона вірна) і визначимо поріг x_α із співвідношення $P(X > x_\alpha) = P(X < x_\alpha) = \frac{\alpha}{2}$, в якому X має стандартний нормальний розподіл.

3. Якщо величина $\hat{X}_s = \frac{X_s - \frac{N}{2}}{\frac{\sqrt{N}}{2}} > x_\alpha$ або, $\hat{X}_s < -x_\alpha$ то H_0^0 відкидається, в іншому випадку – приймається.

Тести d-одночленів

У цьому випадку гіпотеза H_0^d відповідає розподілу . $N\left(\frac{1}{2}\binom{n}{d},\frac{1}{2}\sqrt{\binom{n}{d}}\right)$ Перший тест T_1^d розглядає кожну окрему АНФ і, таким чином, стосується локальної сфери впливу та вирізняє слабкі АНФ. Другий тест T_2^d згруповує N АНФ, утворюючи декілька їх класів. В результаті, буде використовуватись розподіл χ^2 з ν степенями свободи, що відповідає сумі ν квадратів незалежних випадкових величин $\frac{n_d^i - \hat{n}_d^i}{\sqrt{n_d^i}}$ *(i≤ν)*, які за визначенням мають стандартний нормальний розподіл. Для T_1^d *ν=N-1*, для T_2^d *2≤ ν≤9*.

1. для кожного *i*, i≤ *ν*, обчислюємо величини n_d^i та $\hat{n}_d^i$ (n_d^i - очікувана частота, $\hat{n}_d^i$ - величина, що спостерігається).

2. Зафіксуємо рівень значущості α та порогове значення x_α, яке обчислюється з розподілу χ^2 з ν степенями свободи.

3. Обчислюємо статистику D^2 за виразом $D^2 = \sum_{i=1}^{\nu} \frac{\left(n_d^i - \hat{n}_d^i\right)^2}{n_d^i}$.

4. Якщо $D^2 > x_\alpha$, то гіпотеза H_0^d відкидається, в іншому випадку вона приймається.

Тест T_2^d призначений для аналізу розглядуваної криптосистеми з глобальної точки зору. Зокрема, він спрямований на перевірку того, що виявлені тестом T_1^d локальні відхилення все ще є значущими на більш глобальному рівні. Замість того, щоб мати справу з

частотами $\hat{n}_d^i$, які спостерігаються, для кожної АНФ, нас більше цікавить кількість тих АНФ, для яких $\hat{n}_d$ належить наперед визначеному інтервалу *(a,b)*.

Статистичне тестування з використанням векторів ініціалізації

Нехай $K = (k_0, \ldots, k_{N-1})$, позначає секретний ключ адитивного синхронного потокового шифру, $IV=(iv_0, \ldots iv_{m-1})$ позначає відкрите значення вектора ініціалізації *IV*, $Z=z_0,z_1,\ldots$- послідовність ключового потоку. Будемо припускати, що нападник отримав деяку кількість різних послідовностей ключового потоку, згенерованих за допомогою використання різних (можливо, вибраних) значень *IV*.

Різноманітні тести використовувались для оцінювання статистичних властивостей послідовностей, вироблених симетричними шифрами або геш-функціями. Ці тести, зазвичай, беруть одну довгу послідовність ключового потоку, а потім застосовують різні тести (наприклад, комплект тестів, використаних *NIST* для оцінювання *AES*).

Однак, останнім часом декілька дослідників помітили можливість використання множини коротких послідовностей ключового потоку, породжених різними вибраними значеннями *IV*, і дослідження статистичних властивостей лише першого символу на виході кожного ключового потоку. На основі роботи [1], що розглянута вище, Саарінен в своїй роботі [2] запропонував розрізнювач *IV* на основі тесту *d*-одночленів. Поведінка ключового потоку аналізується за допомогою використання функції від *n* змінних *IV*, тобто $z=f(iv_0, \ldots, iv_{n-1})$. Усі інші біти *IV* та біти ключа вважаються константами.

В роботі [3] автори запропонували більш узагальнений підхід. Замість того, щоб аналізувати лише одну функцію, вивчається поведінка більшої кількості многочленів з тим, щоб більш (або менш) ймовірні порівняно з іншими одночлени можна було б виявити. Виберемо *n* значень *IV*, позначених $iv_0, \ldots, iv_{n-1}$, як змінні. Решта значень IV , а також біти ключа є константами. Використовуючи перший біт на виході $z_0=f_1(iv_0, \ldots, iv_{n-1})$. для кожного вибору $iv_0, \ldots, iv_{n-1}$, ми можемо побудувати АНФ функції f_1.

Новий підхід полягає в тому, що повторюється теж саме декілька разів, але при цьому використовуються інші значення *IV*, що не вважаються змінними. Перебравши усі значення $iv_0, \ldots, iv_{n-1}$ у цьому випадку отримаємо функцію f_2 . Продовжуючи у такий спосіб, можна вивести *P* АНФ для різних функцій $f_1, f_2, \ldots, f_P$. У деяких випадках є можливість отримати многочлени, коли використовуються ті ж самі *IV*, але різні ключі. Маючи в своєму розпорядженні різні многочлени, можна будувати будь які тести на їх основі. Тест *d*-одночленів, розглянутий Сааріненом в [2], є випадком, коли $P=1$.

Тест розподілу одночленів

Сценарій атаки є подібним до тесту *d*-одночленів, але замість того, щоб підраховувати кількість одночленів заданого степеня, генерується *P* многочленів і обчислюється, в скількох многочленах присутній кожний одночлен, тобто кількість многочленів, в яких $a_u=1$, $u \in GF(2^n)$. Як еквівалент, можна розглядати $a_i=1$, $0 \le i \le 2^n-1$. Позначимо через M_{a_i} кількість появ таких коефіцієнтів. Тоді M_a є біноміально розподіленою величиною $B\left(P, \frac{1}{2}\right)$. Тест будується на основі статистики

$$\chi^2 = \sum_{i=0}^{2^n-1} \frac{\left(M_{a_i} - \frac{P}{2}\right)^2}{\frac{P}{2}}.$$

Реалізація цього тесту має більшу обчислювальну складність, ніж для тесту *d*-одночленів, а саме $O(P \times n2^n)$ і вимагає наявності $O(n2^n)$ пам'яті. З іншого боку, якщо для шифру деякі одночлени є сильно нерівномірно розподіленими, атака може бути успішною з меншою кількістю бітів *IV*, тобто меншим *n*, порівняно з тестом *d*-одночленів. Додатково, хоч ця атака спочатку пропонувалась для сценарію вибраних *IV* з фіксованим невідомим ключем, є також можливість застосування тесту для різних значень ключа, розглядаючи при цьому одні й ті ж самі біти *IV*.

Одночлен максимального степеня

Дуже простим тестом є спостерігання одночлена максимального степеня, який може бути вироблений генератором ключового потоку. Одночлен максимального степеня є добутком усіх бітів *IV* і, отже, може з'явитись лише, якщо усі біти *IV* добре змішані. Тест, подібно попереднім тестам, здійснюється шляхом ініціалізації шифру усіма можливими комбінаціями *n* бітів *IV* $z^{iv_0,\ldots,iv_{n-1}} = f(iv_0,\ldots,iv_{n-1})$ при фіксованих інших бітах. Існування одночлена максимального степеня можна перевірити за допомогою операції *XOR*, яка застосовується до першого біта ключового потоку з кожної ініціалізації, так як

$$a_{2^n-1} = \bigoplus_{iv_0,\ldots,iv_{n-1}} z^{iv_0,\ldots,iv_{n-1}}$$

Змінюючи деякі інші біти *IV*, можна отримати новий многочлен, здійснити цю процедуру для *P* многочленів і спостерігати, чи зустрічається одночлен максимального степеня в усіх многочленах або не зустрічається в жодному. У такому випадку функція відрізняється від випадкової. Отже, можна з низькою обчислювальною складністю і майже без елементів пам'яті перевірити, чи може існувати одночлен максимального степеня на виході шифру.

Застосування АНФ для знаходження ключів

Ідеї методів, описаних вище, були застосовані в роботі [4] для знаходження ключів Нехай задано деяку фіксовану булеву функцію $F(K,V): \{0,1\}^n \times \{0,1\}^m \to \{0,1\}$. Оракул вибирає випадковий і невідомий $K = (k_0,\ldots,k_{n-1})$ та повертає значення $z = F(K,V)$ для кожного запиту $V = (v_0,\ldots,v_{m-1})$, який можна вибирати. Функція F може позначати, наприклад, булеву функцію, яка відображає ключ K і значення V вектора ініціалізації потокового шифру в (скажімо) перший біт на виході. Метою нападника є визначення невідомого ключа K (або розрізнити F від випадкової функції) в моделі атаки з вибраним *IV*, маючи справу лише з функцією F. Якщо F змішує вхідні дані належним чином, то потрібно перебрати всі можливі 2^n ключів, посилаючи $O(n)$ запитів до оракула для того, щоб знайти вірний ключ (так як кожний запит дає один біт інформації про ключ для збалансованої F). Спробуємо дослідити методи, що потенційно ведуть до більш швидкого знаходження ключа

у випадку, коли функція F змішує свої вхідні дані неналежним чином. Це може трапитись, наприклад, коли фаза ініціалізації потокового шифру здійснюється за допомогою ітераційної процедури, для якої число ітерацій вибрано в не найкращий спосіб. З іншого боку, ці методи можуть допомогти винахідникам шифрів більш прискіпливо вибирати необхідне число ітерацій. Існування більш швидких методів знаходження невідомого ключа K в значній мірі залежить від структури функції K. Навіть може виявитись неможливим однозначно визначити ключ K. Нехай $F(K,V)=\oplus_u C_u(V)K^u$, де $K^u=k_0^{u_0}\ldots k_0^{u_{n-1}}$ для мультиіндексу $u=(u_0,\ldots,u_{n-1})$ Тоді має місце наступне.

Твердження [4]. Жоден нападник неспроможний розрізнити два ключі K_1 та K_2, для яких $k_1^u=k_2^u$ для усіх $u\in\{0,1\}^n$ таких, що $C_u(V)\neq 0$.

Наслідком цього твердження є те, що функція F ділить $\{0,1\}^n$ на класи еквівалентності $K_1,\ldots,K_J$ (з $J\leq 2^n$).

Алгебраїчний опис функції $F(K,V)$ у загальному випадку є досить складним для безпосереднього аналізу. Тому, виходячи з функції $F(K,V)$ і розбиття $V=(U,W)$, можна вивести простіші булеві функції $C(K,W)$ за допомогою оракула. Наприклад, $C(K,W)$може бути коефіцієнтом АНФ функції, виведеної з F ,коли змінюються лише біти з U. Якщо ця функція $C(K,W)$ не має добре розподіленої алгебраїчної структури, її можна експлуатувати в криптоаналітичних атаках. Дослідимо різні сценарії:

1. Якщо $C(K,W)$ незбалансована для (необов'язково рівномірно розподіленого) випадкового W і багатьох фіксованих K , то функцію F (або еквівалентний досліджуваний потоковий шифр) з невідомим K можна відрізнити від випадкової функції.

2. Якщо обчислити $C(K,W)$ для деякого фіксованого W , то $C(K,W)$ є виразом лише від бітів ключа. В [5] показано, що у випадку шифру *Trivium* з меншим числом ітерацій можна вивести лінійні співвідношення для бітів ключа, якщо вдало вибрати частину *IV*.

3. Якщо $C(K,W)$ має багато бітів ключа, які (майже) не впливають на значення $C(K,W)$, можна знайти прийнятну апроксимацію і використовувати її в атаках розкриття ключа.

В сценарії 2 основоположною ідеєю є знаходження співвідношення $C(K,W)$, яке обчислюється для деякого W, що залежить лише від підмножини з t $(<n)$ бітів ключа. Функціональну форму цього співвідношення можна визначити за допомогою 2^t обчислень функції $C(K,W)$. Перебираючи усі 2^t можливостей для задіяних t бітів ключа, можна відфільтрувати ті ключі, які не задовольняють співвідношенню. Складністю цього передобчислення є 2^t кроків, необхідних для обчислень $C(K,W)$. Більш точно, якщо $p=P(C(K,W)=0)$ для фіксованого W, то ключовий простір відфільтрується на множник $H(p)=p^2+(1-p)^2$. Наприклад, у випадку лінійної функції $p=H(p)=1/2$. В доповнення, якщо доступні декілька співвідношень для бітів ключа, то їх легше скомпонувати для того, щоб відфільтрувати невірні ключі, якщо вони мають просту структуру. В сценарії 3 основною ідеєю є знаходження функції $A(L,W)$, яка залежить від частини ключа L, що складається з t бітів, і яка корельована з $C(K,W)$ з коефіцієнтом кореляції ε, тобто $P\{C(K,W)=A(L,W)\}=\frac{1}{2}(1+\varepsilon)$. Тоді, подаючи оракулу N запитів, можна отримати деяку

інформацію (залежно від нових класів еквівалентності, породжених функцією A) про t бітів секретного ключа K за час $N2^t$, уважно аналізуючи проблему перевірки гіпотези.

Функцію F можна записати у вигляді $F(K,V)=\oplus_{v,u} C_{v,u} V^v K^u$ з бінарними коефіцієнтами $C_{v,u}$. Можна розбити IV у відповідності з $V=(U,W)$ і $v=(\alpha,\beta)$ на l-бітові сегменти U і α та $(m\text{-}l)$-бітові сегменти W і β. Отримаємо вираз $F(K,V)=\oplus_{\alpha,\beta,u} C_{(\alpha,\beta),u} U^\alpha W^\beta K^u = \oplus_\alpha C_\alpha(K,W) U^\alpha$, де $C_\alpha(K,W)=\oplus_{\beta,u} C_{(\alpha,\beta),u} W^\beta K^u$. Для кожного $\alpha \in \{0,1\}^l$ функція $C_\alpha(K,W)$ може слугувати за функцію C, виведену з F. Для ілюстрації розглянемо іграшковий приклад.

Приклад 1. Нехай $n=m=3$, а $F(K,W)=k_1 v_1 \oplus k_2 v_0 v_2 \oplus v_2$. Нехай $U:=(v_0,v_2)$ має l бітів, а $W:=(v_1)$ має $m\text{-}l=1$ біт. Тоді $C_0(K,W)=k_1 v_1$, $C_1(K,W)=\mathbf{0}$, $C_2(K,W)=\mathbf{1}$, $C_3(K,W)=k_2$.

Відзначимо, що нападник за допомогою оракула може обчислити $C_\alpha(K,W)$ для невідомого ключа K при будь-якому $W \in \{0,1\}^{m-l}$ для кожного $\alpha \in \{0,1\}^l$, посилаючи щонайбільше 2^l запитів до оракула. Іншими словами, розбиття V допомагає визначити обчислювальну функцію $C_\alpha(K,W)$ для малих значень l навіть тоді, коли точна форма функції $C_\alpha(K,W)$ залишається невідомою. Для того, щоб отримати значення $C_\alpha(K,W)$ для усіх $\alpha \in \{0,1\}^l$, нападник отримує значення на виході для усіх 2^l вхідних даних $V=(U,W)$ з фіксованою частиною W. Це дає таблицю істинності булевої функції від l змінних, для якої коефіцієнти її АНФ (тобто значення величин функції $C_\alpha(K,W)$) можна знайти за час $l2^l$, використовуючи при цьому 2^l комірок пам'яті.

Можна очікувати, що підмножина бітів IV має менше змішування в процесі ініціалізації, ніж інші біти. Ці біти IV називають слабкими і вони повинні бути відповідним вибором U для того, щоб посилити нерівномірність C.

Нас цікавить апроксимація заданої функції $C(K,V): \{0,1\}^n \times \{0,1\}^{m-1} \to \{0,1\}$, яка залежить лише від підмножини бітів ключа. З цією метою роблять відповідне розбиття ключа $K=(L,M)$, в якому L містить t значущих бітів ключа, а M містить остачу $n\text{-}t$ незначущих бітів ключа, і будують функцію $A(L,W)$. Далі використовується термін *підключ*, який стосується множини значущих бітів ключа. Таке розбиття можна здійснювати у систематичний спосіб, використовуючи поняття ймовірнісних нейтральних бітів, уведеного в роботі [6].

Визначення 1. Міра нейтральності біта k_i ключа відносно функції $C(K,W)$ визначається як γ_i, де $\frac{1}{2}(1+\gamma)$ є ймовірністю (за всіма K та W) того що інвертування біта k_i ключа не змінює вихідні дані функції $C(K,W)$.

На практиці встановлюють поріг γ такий, що усі біта ключа з $|\gamma_i| > \gamma$ включаються у підключ L (тобто ймовірнісні нейтральні біти ключа вибираються згідно з індивідуальними значеннями їх міри нейтральності). Апроксимація $A(L,W)$ визначається функцією $C(K,W)$, в якій незначущі M бітів ключа покладаються рівними нулеві.

Приклад 2. Нехай $n=m=3$, $l=2, C(K,V)=k_0 k_1 k_2 v_0 v_1 \oplus k_0 v_1 \oplus k_1 v_0$. Для рівномірно розподілених випадкових K і W $\gamma_0 = 1/8, \gamma_1 = 1/8, \gamma_2 = 7/8$ Отже, має сенс використовувати $L:=(k_0,k_1)$ як підключ. З фіксованим $k_2=0$ отримують апроксимацію $A(L,V)=k_0 v_1 \oplus k_1 v_0$, яка залежить лише від $t=2$ бітів ключа. Відзначимо,

якщо M складається лише з нейтральних бітів ключа (з γ_i=1), то апроксимація A є точною, тому що $C(K,W)$ не залежить від цих бітів ключа.

У фазі передобчислень атаки необхідно здійснити відповідне розбиття IV і ключа (щоб отримати функцію C і апроксимацію A). Слабкі біти IV часто знаходять за допомогою випадкового пошуку, в той час як слабкі біти ключа можна легко знайти за допомогою міри нейтральності для деякого порогу γ. Для заданих C і A можна знайти малу підмножину кандидатів для підключа L за допомогою ймовірнісної атаки «вгадуй і визначай». Для того, щоб відфільтрувати множину усіх можливих підключів в меншу множину, необхідно розрізнити вірне вгадування підключа $\hat{L}$ від невірного вгадування. Спроможність розрізнювання підключів пов'язана з коефіцієнтами кореляції між $A(\hat{L},W)$ і $C(K,W)$ при K=(L,M) та при таких двох гіпотезах H_0: вгадувана частина $\hat{L}$ - вірна і H_1: вгадувана частина $\hat{L}$ - невірна. Більш точно, визначальну роль відіграють величини ε_0 і ε_1, що визначаються наступним чином:

$$P_W\{A(\hat{L},W)=C(K,W)\mid K=(\hat{L},M)\}=\frac{1}{2}(1+\varepsilon_0)$$

$$P_{\hat{L},W}\{A(\hat{L},W)=C(K,W)\mid K=(L,M)\}=\frac{1}{2}(1+\varepsilon_1)$$

У загальному випадку ε_0 і ε_1 - випадкові величини, що залежать від ключа. У випадку, коли розподіли ε_0 та ε_1 розрізняються, можна досягти малої ймовірності не виявлення p_{mis} і ймовірності хибної тревоги p_{fa}, використовуючи достатню кількість вибіркових значень. У випадку, коли ε_0 та ε_1 - константи з $\varepsilon_0>\varepsilon_1$, оптимальним є критерій Неймана-Пірсона. Тоді N значень $C(K,W)$ для різних W (припускаючи, що вибіркові значення $C(K,W)$ є незалежними) достатньо, щоб отримати $p_{fa}=2^{-c}$ і $p_{mis}=1{,}3*10^{-3}$, де

$$N\approx\left(\frac{\sqrt{2c(1-\varepsilon_0^2)\ln 2}+3\sqrt{1-\varepsilon_1^2}}{\varepsilon_1-\varepsilon_0}\right)^2.$$

Атака буде успішною з ймовірністю 1-p_{mis} і такої складності. Для кожного вгадування $\hat{L}$ підключа повинна бути обчислена кореляція ε величини $A(\hat{L},W)\oplus C(K,W)$, що потребує обчислення нападником коефіцієнтів $A(\hat{L},W)$ і $C(K,W)$ за допомогою оракула для N значень W щонайбільше за $N2^t$ кроків. Це потрібно повторити для усіх 2^t можливих вгадувань $\hat{L}$. Множина кандидатів для підключа L має розмір біля $p_{fa}2^t=2^{t-c}$. Увесь ключ можна перевірити шляхом повного перебору частини ключа M коштом $2^{t-c}*2^{n-t}$ обчислень функції F. Загальна складність стає рівною $N2^l2^t+2^{t-c}*2^{n-t}=N2^{l+1}2^{n-c}$. Використання більш ніж однієї функції C або розгляд декількох вибраних бітів U вектора ініціалізації може бути корисним для зниження складності.

Зауваження 1. На практиці величини ε_0 та ε_1 залежать від ключа. Якщо розглядати ключ як випадкову величину, то ε_0 та ε_1 є також випадковими величинами. Однак, їх розподіли можуть бути неповністю розрізнювані, а отже можна не досягти малих значень p_{mis} та p_{fa}. Пропонується такий неоптимальний розрізнювач: спочатку вибирається поріг ε_0^* такий, що $p_\varepsilon=P\{\varepsilon_0>\varepsilon_0^*\}$ має значну величину, наприклад, 1/2. Якщо є можливість, то визначається поріг ε_1^* такий, що $P\{\varepsilon_1<\varepsilon_1^*\}=1$. Потім оцінюється розмір вибірки, замінюючи ε_0 та ε_1 відповідно ε_0^* та ε_1^*, щоб отримати $p_{fa}\le 2^{-c}$ і $p_{mis}\cdot p_c\approx\frac{1}{2}$. Якщо ε_0^* та ε_1^* близькі

одне до одного, то оцінюване число вибіркових значень стає дуже великим. У цьому випадку краще інтуїтивно вибрати число вибіркових значень, а потім оцінити p_{fa}..

Зауваження 2. Має сенс припустити, що хибний підключ $\hat{L}$,який є близьким до вірного ключа, може привести до більшого значення ε. Мірою «близькості» можуть слугувати міра нейтральності γ_i та вага Хемінга: якщо лише декілька бітів ключа з великими γ_i є хибними, то можна очікувати, що ε велике.

Розв'язування нелінійних рівнянь з використанням АНФ та його застосування у криптоаналізі потокових шифрів

Для розв'язування нелінійних рівнянь з багатьма змінними було запропоновано декілька методів [7-9]. Вони отримали новий імпульс для свого розвитку з часу винайдення так званих алгебраїчних атак на криптосистеми [10-11]. Далі розглядається підхід, запропонований в роботі [12]. Основним спостереженням, що використовується в цій роботі, є те, що поліноміальні рівняння, які визначаються багатьма криптосистемами не є довільними і не зв'язаними одне з одним. Замість цього, вони є новими варіантами, виведеними з єдиного головного многочлена шляхом призначення деяким змінним значень, що визначаються нападником. В потокових шифрах, як вже було зазначено вище, вихідні дані залежать від фіксованих бітів секретного ключа і бітів відкритого *IV*, останні з яких можна довільно вибирати. Змінюючи значення цих відкритих бітів, нападник може отримати багато виведених поліноміальних рівнянь, які тісно пов'язані одне з одним. Нижче буде показано, якщо головний многочлен є достатньо випадковим, то з великою ймовірністю можна вилучити усі його n^2 нелінійних членів, розглядаючи напрочуд малу кількість лише $2^d n$ варіантів, а потім розв'язати передобчислену версію n лінійних рівнянь з n змінними, використовуючи лише n^2 бітових операцій. Наприклад, якщо d=16 і n=10000, одночасно можна вилучити усі 2^{200} нелінійних термів, розглядаючи лише 2^{20} виведених поліноміальних рівнянь. Після вилучення цих нелінійних термів, єдине, що залишається, це – система лінійних рівнянь від усіх секретних бітів, яку легко розв'язати.

Для ілюстрації розглянемо досить щільний головний многочлен степеня d=3 від трьох секретних змінних x_1, x_2, x_3 і трьох відкритих змінних v_1, v_2, v_3:

$$P(v_1, v_2, v_3, x_1, x_2, x_3) = \oplus v_1 v_2 v_3 \oplus v_1 v_2 x_1 \oplus v_1 v_2 x_1 \oplus v_2 v_3 x_1 \oplus v_1 v_2 x_3 \oplus v_2 v_3 x_2 \oplus v_1 v_3 x_3 \oplus$$
$$\oplus v_1 x_1 x_3 \oplus v_3 x_2 x_3 \oplus x_1 x_2 x_3 \oplus v_1 v_2 \oplus v_1 x_3 \oplus v_3 x_1 \oplus x_1 x_2 \oplus x_2 x_3 \oplus x_2 \oplus v_1 \oplus v_3 \oplus \mathbf{1}$$

Многочлени третього степеня від шести змінних можуть мати $\binom{6}{3}+\binom{6}{2}+\binom{6}{1}+\binom{6}{0} = \mathbf{42}$ можливих термів і отже існує 2^{42} таких многочленів над полем *GF*(2). Щоб вилучити усі 35 можливих нелінійних термів за допомогою гаусівського вилучення, потрібно 35 таких многочленів. Покладаючи три відкриті змінні v_1, v_2, v_3 усім можливим 0/1 значенням, можна отримати лише 8 виведених многочленів,що може виявитись недостатнім. Однак, підсумовуючи 4 виведені многочлени з v_1=0,отримуємо $x_1 \oplus x_2$, підсумовуючи 4 виведені многочлени з v_2=0, отримуємо, $x_1 \oplus x_2 \oplus x_3$ підсумовуючи 4 виведені многочлени з v_3=0, отримуємо $x_1 \oplus x_2$, які одночасно вилучають усі нелінійні члени. Якщо чисельно за модулем 2 підсумувати значення виведених многочленів у ці три різні способи (замість посимвольного підсумовування самих многочленів), отримуємо просту систему трьох лінійних рівнянь від трьох секретних змінних. Отже, головний нелінійний многочлен може бути розв'язаний за допомогою атаки з вибраними відкритими даними, яка обчислює його лише для 8 комбінацій значень його відкритих змінних.

Так як доводиться мати справу зі щільними многочленами від багатьох змінних відносно великого порядку, їх точне представлення є надзвичайно великим і тому

припускається, що вони надаються лише неявно як чорні ящики, яким можна надсилати запити. Це є природнім припущенням у криптоаналізі, в якому нападник може взаємодіяти з чорним ящиком шифрування, який містить секретний ключ. Результатом такого підходу може бути наявність атак на повністю невідомі криптосистеми, які вкладені в інтегральні схеми, захищені від фізичного доступу, без досить коштовного процесу оберненого інженірінгу.

Серед питань, які треба розглядати у даному підході, є те , як ефективно оцінити степінь d многочлена з багатьма змінними, заданого чорним ящиком, як розв'язати многочлени з великими степенями, які можуть бути апроксимовані многочленами з малими степенями, як легко знайти лінійні рівняння, визначені сумами їх величезних виведених многочленів. Відзначимо, що в моделі чорного ящика нападнику не дозволяється здійснювати символьні операції, такі як запитування коефіцієнта конкретного терма, обчислення НСД двох многочленів або обчислення базису Гробнера до тих пір, поки він інтерполює їх з їх значень за допомогою трудомісткої процедури, яка вимагає величезної кількості запитів.

Запропонований в [12] метод названо *кубічною атакою*, так як в ній деякі відкриті змінні набувають усі можливі значення в n (d-1)-мірних булевих кубах і підсумовуються результати в кожному кубі. Ця атака не є повністю новою, так як деякі з ідей і методів вирисовувались в попередніх евристичних атаках на різні криптосистеми, але в ній вперше усі елементи зібрані разом та супроводжуються уважним аналізом складності та ймовірності успіху для випадкових многочленів чорного ящика.

Нападнику надається чорний ящик, який обчислює невідомий многочлен p над полем $GF(2)$ від $n+m$ вхідних бітів $(x_1,\ldots,x_n,v_1,\ldots,v_m)$ і видає один біт на виході. Вхідні біти $x_1,\cdots,x_n$ є секретними змінними, в той час як $v_1,\ldots,v_m$ є відкритими змінними. Розв'язання складається з двох фаз. Під час першої фази передобчислень нападнику дозволяється надавати значення усім змінним $(x_1,\ldots,x_n,v_1,\ldots,v_m)$ і використовувати чорний ящик для того, щоб обчислювати відповідний біт на виході p. Під час другої фази n секретних змінних приймають невідомі значення, а нападнику дозволяється встановлювати значення m відкритих змінних $(v_1,\ldots,v_m)$ за своїм бажанням і обчислювати вихідні значення p.

Далі для спрощення позначень не розрізняються секретні та відкриті змінні і позначаються усі як $x_1,\cdots,x_n$. Так як $x_i^2=x_i$ за модулем 2, то терми t_I у многочлені можна індексувати підмножиною $I\subseteq\{1,\cdots,n\}$ змінних, які перемножуються, і многочлени можуть бути зображені як суми t_I для різних підмножин I. Позначатимемо P_d^n множину усіх многочленів над полем $GF(2)$ від n змінних, степені яких не перевищують d.

При заданих многочлені з багатьма змінними p і підмножині індексів I можна виділити спільний підтерм t_I з деяких термів многочлена p і представити многочлен як суму термів, які є надмножинами I та термів, які не є надмножинами I:

$$p(x_1,\ldots,x_n)\equiv t_I\cdot P_{S(I)}+q(x_1,\ldots,x_n).$$

$P_{S(I)}$ називатимемо надмногочленом I в многочлені p. Відзначимо, що для будь-яких p та I, надмногочлен I в многочлені p є многочленом, який не містить спільних з t_i змінних, а в кожному термі многочлена $q(x_1,\ldots,x_n)$ відсутня хоча б одна змінна з I.

Для ілюстрації уведених понять розглянемо

$$p(x_1,\ldots,x_n)=x_1x_2x_3\oplus x_1x_2x_4\oplus x_2x_4x_5\oplus x_1x_2\oplus x_2\oplus x_3x_5\oplus x_5\oplus 1,$$

що є многочленом степеня 3 від п'яти змінних. Нехай $I=\{1,2\}$ є підмножиною індексів розмірності 2. Многочлен p можна представити як

$$p(x_1,\ldots,x_n) = t_I \cdot P_{S(I)} \oplus q(x_1,\ldots,x_n)$$

$$p(x_1,x_2,x_3,x_4,x_5) = x_1x_2(x_3 \oplus x_4 \oplus \mathbf{1}) \oplus (x_2x_4x_5 \oplus x_3x_5 \oplus x_2 \oplus x_5 \oplus \mathbf{1}),$$

в якому

$$t_i = x_1x_2,$$

$$P_{S(I)} = x_3 \oplus x_4 \oplus \mathbf{1},$$

$$q(x_1,x_2,x_3,x_4,x_5) = (x_2x_4x_5 \oplus x_3x_5 \oplus x_2 \oplus x_5 \oplus \mathbf{1})$$

Визначення 2. Макстермом многочлена p є терм t_I такий що $\deg(P_{S(I)}) \equiv \mathbf{1}$,тобто надмногочлен I у многочлені p є лінійним многочленом, який не є константою.

Будь яка підмножина I розмірності k визначає k-вимірний булевий куб C_I 2^k булевих векторів, в якому призначаються усі можливі комбінації 0/1 значень змінним з I, а усі останні змінні залишаються невизначеними. Будь-який вектор $v \in C_I$ визначає новий виведений многочлен $p|_v$ від n-k змінних. Підсумовуючи ці виведені многочлени для усіх 2^k можливих векторів з C_I, отримуємо в результаті новий многочлен, який позначається як $P_I = \sum_{v \in C_I} p|_v$.

Має місце наступна

Теорема 1 [12]. Для будь якого многочлена p і підмножини змінних I $P_I \equiv P_{S(I)} (\mathrm{mod}\, \mathbf{2})$.

З теореми випливає, що для того, щоб розв'язати головний многочлен степеня d, нападнику достатньо знайти в ньому багато макстермів і для кожного макстерма підсумувати щонайбільше 2^{d-1} виведених многочленів. Відзначимо, що він має додавати лише 0/1 значення цих виведених многочленів, а не їх величезні символьні вирази. Отриманий в результаті біт прирівнюється до фіксованого лінійного виразу, який можна отримати з головного многочлена чорного ящика під час окремої фази передобчислення, так як він не є залежним від ключа. Для невеликих степенів (як от d=16) виведення правої частини кожного рівняння під час другої фази вимагає щонайбільше 2^{15} додавань єдиного біта на виході, що займає дуже мало часу.

Фаза передобчислення.

Якщо задано точний опис головного многочлена, то легко представити його у вигляді $p(x_1,\ldots,x_n) = t_I \cdot P_{S(I)} \oplus q(x_1,\ldots,x_n)$ для будь-якого терму t_I . Однак, коли експоненціально довгий головний многочлен задано як чорний ящик, не зрозуміло, як знайти його представлення і як зберігати його у компактному вигляді. Якщо t_I є макстермом, питання компактного зображення легко вирішується, так як потрібно лише знати його надмногочлен $P_{S(I)}$ для здійснення атаки, а цей вираз є короткою лінійною комбінацією деяких секретних змінних x_i, до яких може додаватись одиниця. Відзначимо, що можна вилучити усі відкриті змінні v_i, які не приймають участь у підсумовуванні, з цього лінійного виразу, поклавши кожну з них рівною 0 (або 1) під час підсумовування.

Для того, щоб знайти $P_{S(I)}$ для заданого чорним ящиком многочлена і макстерм t_I в ньому, використовується окрема фаза передобчислень, в якій нападнику надається можливість вибирати значення як відкритих, так і секретних змінних.

Теорема 2. [12] Нехай t_I є макстермом в многочлені чорного ящика p. Тоді:

1. Вільний терм в $P_{S(I)}$ можна обчислити, підсумовуючи за модулем 2 значення многочлена p для усіх вхідних n+m змінних, які дорівнюють нулеві всюди за виключенням d-1 змінних в кубі C_I.

2. Коефіцієнт при x_i у лінійному виразі $P_{S(I)}$ можна обчислити, підсумовуючи за модулем 2 значення многочлена p для усіх вхідних векторів, які рівні нулеві всюди за

виключенням куба C_I і значення многочлена p для усіх вхідних векторів за виключенням куба C_I та змінної x_i, яка покладається рівною одиниці.

Передобчислення для випадкових многочленів

У багатьох криптографічних схемах змішування вхідних даних є настільки інтенсивним, що многочлен, яким представляється кожен біт шифрованого тексту, можна вважати випадковим.

Визначення 3. d–випадковим многочленом степеня d від $n+m$ змінних є многочлен $p \in P_d^{n+m}$ такий, що кожний можливий терм степеня d, який містить одну секретну змінну і d-1 відкритих змінних, вибирається незалежно з ймовірністю 0,5, а усі інші терми можуть вибиратись довільним чином.

У будь якому d-випадковому многочленів кожен терм, який є добутком v-1 відкритих змінних v_i має надзвичайно велику ймовірність бути макстермом. Відповідним надмногочленом є многочлен степеня щонайбільше 1 і він є многочленом степеня 0 лише тоді, коли для усіх секретних змінних x_i терми t_I не вибираються у цьому многочлені. Ймовірність такої події дорівнює 2^{-n}.

Після вибору n випадкових макстермів нападник визначає $n \times n$ матрицю A, рядки якої містять їх відповідні надмногочлени. Якщо матриця невироджена, то нападник обчислює і зберігає A^{-1} для того, щоб зменшити складність розв'язування систем лінійних рівнянь у другій фазі з $O(n^3)$ до $O(n^2)$.

Передобчислення для невипадкових многочленів

Для будь якого вибору значень для усіх секретних змінних підсумовуємо 0/1 значення многочлена p для усіх відкритих значень куба C_I, поклавши усі інші відкриті змінні рівними 0. Ця сума є функцією лише від секретних змінних і можна перевірити її лінійність, використовуючи один з ефективних тестів лінійності [13].

Прикладом такого тесту лінійності є *BLR*-тест [14], який вибирає вектори $x, y \in \{\mathbf{0},\mathbf{1}\}^n$ випадково та незалежно і перевіряє співвідношення

$$P_{S(I)}[\mathbf{0}] \oplus P_{S(I)}[x] \oplus P_{S(I)}[y] = P_{S(I)}[x \oplus y].$$

Тест забезпечує те, що якщо $P_{S(I)}$ є лінійним, то тест завжди досягає успіху, а якщо $P_{S(I)}$ далекий від лінійного, то тест з великою ймовірністю зазнає невдачі. Тест повторюють достатнє число разів, доки нападник не впевниться, що $P_{S(I)}$ є дуже близьким до лінійного (тобто він є лінійним за виключенням декількох термів вищих ступенів, які майже завжди приймають нульове значення). Використовуючи кубічну атаку у цьому випадку, можна знайти більшість, але не усі можливі ключі, що теж можна вважати успіхом криптоаналізу. Як і для випадкових многочленів, нападник зупиняється, коли виведено достатньо багато лінійно-незалежних векторів і можна обчислити A^{-1}. Друга фаза атаки така ж, як і для випадкових многочленів.

Застосування до потокових шифрів

В моделі припускається, що нападник може моделювати шифр під час фази передобчислень і може застосувати атаку з вибраним *IV* під час другої фази. Багато запропонованих шифрів використовують один або більше лінійних регістрів зсуву (ЛРЗ), які фільтруються або поєднуються нелінійними функціями для вироблення бітів на виході. Атака вимагає знання ле одного біта на виході для різних значень *IV* і існує можливість вибору місця розташування цього біта. Зокрема, можна вибрати таке місце розташування

біта, в якому відомим є біт відкритого тексту. Типовими прикладами таких місць розташування є біти стандартного заголовка пакета або старші біти символів *ASCII*, які рівні 0. Якщо нападнику доступні більш ніж один біт на виході під час роботи потокового шифру., то він може дещо зменшити кількість відкритих змінних, необхідних для атаки, підсумовуючи вихідні дані декількох многочленів p_i, які визначають біти різних виходів. У цей спосіб він може отримати більш ніж одне лінійне рівняння для кожного макстерму під час фази перед обчислень, що дозволить йому запускати потоковий шифр меншу кількість разів.

Список літератури

1. *E.Filiol.* A new statistical testing for symmetric ciphers and hash functions.// In Cryptology eprint Archive, Report 2002/099.

2. *M.-J.O.Saarinen.* Chosen-IV statistical attacks against eSTREAM ciphers. In SECRYPT 2006.

3. *H.Englund, T.Johansson and M.S.Turan.* A Framework for chosen IV statistical analysis of stream ciphers.// Advances in cryptology. INDOCRYPT 2007, Lecture Notes in Computer Science, Springer-Verlag. – 2007, pages 268-281, vol. 4859.

4. *S.Fischer, S.Kazaei and W.Meier.* Chosen IV statistical analysis for key recovery attacks on stream ciphers.// Advances in cryptology. AFRICACRYPT 2008, Lecture Notes in Computer Science, Springer-Verlag. – 2008, pages 236-245, vol. 5023.

5. *M.Vielhaber.* Breaking ONE. FIVIUM by AIDA an algebraic IV differential attack. In Cryptology ePrint Archive, Report 2007/413.

6. *J.-Ph.Aumasson, S.Fischer, S.Khazaei, W.Meier and C.Rechberger.*New features of Latin dances: analysis of Salsa, ChaCha and Rumba.// Fast Software Encryption Lecture Notes in Computer Science, Springer-Verlag. – 2008, pages 470-488, vol. 5086.

7. *I.A.Ajwa, Z.Lin and P.S.Wang.* Grobner bases algorithm, ICM Technical Report, Feb.1995.

8. *J.-C.Faugire.* A new efficient algorithm for computing Grobner bases (T-4). //Journal of Pure and Applied Algebra.- 1999.- v.139.- P.61-88.

9. *G.Ars, J.-C.Faugire, H.Imai, M.Kavazoe and M.Sugita.* Comparison between XL and Grobner basis algorithm. //Advances in cryptology. ASIACRYPT 2004, Lecture Notes in Computer Science, Springer-Verlag. – 2004, pages 338-353, vol. 3329.

10. *N.Courtois and W.Meier.* Algebraic attacks on stream ciphers with linear feedback. //Advances in cryptology. EUROCRYPT 2003. Lecture Notes in Computer Science, Springer-Verlag. – 2003, pages 346-359, vol. 2656.

11. *N.Courtois.* Fast algebraic attacks on stream ciphers with linear feedback. In Proceedings of CRYPTO 2003, LNCS 2729, 176-194, 2003. Advances in cryptology. CRYPTO 2003. Lecture Notes in Computer Science, Springer-Verlag. – 2003, pages 176-194, vol. 2729.

12. I.*Dinur and A.Shamir.* Cube attacks on tweakable black box polynomials. In Cryptology ePrint Archive, Report 2008/385.

13. *S.Arora and S.Safra.* Probabilistic checking of proofs: A new characterization of NP. //In Proceedings of 33rd Ann. Symp. On Foundations of Computer Science.- 1992.- P.2-13.

14. *M. Blum, M. Luby and R.Rubinfeld.* Self-testing/ correcting with applications to numerical problems. // In Proceedings of 22nd Annual ACM Symp. On Theory of Computing.- 1990.- P. 73-83.

Надійшла 26.12.2008