Розглядається режим вибіркового гамування із прискореним виробленням імітовставки (Galois/Counter Mode and GMAC), специфікацію якого наведено у стандарті NIST SP 800-38D. Розробляється зменшена модель режиму, яка зберігає алгебраїчну структуру всіх основних криптоперетворень та дозволяє за рахунок їхнього масштабування провести експериментальні дослідження колізійних властивостей формованих імітовставок з подальшим прогнозуванням рівня криптографічного стійкості повної версії шифру

Ключові слова: міні-версія режиму шифрування, моделювання, імітовставка, поліноміальне ґешування, блоковий симетричний шифр

Рассматривается режим выборочного гаммирования с ускоренной выработкой имитовставки (Galois/Counter Mode and GMAC), спецификация которого представлена в NIST SP 800-38D. Разрабатывается уменьшенная модель режима, которая сохраняет алгебраическую структуру всех основных криптопреобразований и позволяет за счёт их масштабирования провести экспериментальные исследования коллизионных свойств сформированных имитовставок с последующим прогнозированием уровня криптографической стойкости полной версии шифра

Ключевые слова: мини-версия режима шифрования, моделирование, имитовставка, полиномиальное хеширование, блочный симметричный шифр

УДК 004.056.55
DOI: 10.15587/1729-4061.2014.27888

МОДЕЛЮВАННЯ РЕЖИМУ ВИБІРКОВОГО ГАМУВАННЯ ІЗ ПРИСКОРЕНИМ ВИРОБЛЕННЯМ ІМІТОВСТАВКИ

О. О. Кузнецов
Доктор технічних наук, професор
Кафедра безпеки інформаційних систем та технологій
Харківський національний університет ім. В. Н. Каразіна
пл. Свободи, 4, м. Харків, Україна, 61022
E-mail: kuznetsov_alex@rambler.ru
Є. П. Колованова
Старший викладач*
E-mail: e.kolovanova@gmail.com
Д. В. Іваненко
Кандидат технічних наук, старший викладач*
E-mail: i8o.dima@gmail.com
О. А. Винокурова
Доктор технічних наук, професор*
E-mail: vinokurova@kture.kharkov.ua
*Кафедра безпеки інформаційних технологій
Харківський національний університет радіоелектроніки
пр. Леніна, 14, м. Харків, Україна, 61166

1. Вступ

Для забезпечення безпеки інформаційних та комунікаційних систем зазвичай застосовуються різні механізми захисту, зокрема блокове симетричне шифрування. Його сутність полягає у перетворенні відкритої інформації з використанням спеціальних ключових даних з метою приховування інформаційного змісту повідомлення, підтвердження його справжності, цілісності, авторства, тощо [1–3]. Втім ефективність криптографічного захисту інформації визначається не лише властивостями блокового симетричного шифру, але і способами його використання, тобто режимом застосування симетричних криптоперетворень [1–4].

Для забезпечення цілісності та конфіденційності інформації призначено режим вибіркового гамування із прискореним виробленням імітовставки, що за міжнародною термінологією позначається як Galois/Counter Mode and GMAC (GCM&GMAC) та який специфіковано у стандарті NIST SP 800-38D [4]. Цей режим призначено для реалізації швидкого криптоперетворення при забезпеченні послуг безпеки інформації із використанням різних криптографічних примітивів, зокрема поліноміального ґешування, гамування, тощо [8–25]. Імплементація положень режиму GCM&GMAC для застосування в Україні, його дослідження та аналіз певних властивостей є надзвичайно важливою та актуальною задачею, яка тісно пов'язана із розробкою проекту національного стандарту ДСТУ «Інформаційні технології. Криптографічний захист інформації. Алгоритм симетричного блокового перетворення» [5, 6].

2. Аналіз літературних джерел і постановка проблеми дослідження

Режим GCM&GMAC був стандартизований національним інститутом стандартів і технологій (NIST) [4, 16] та призначений для використання в поєднанні з 128-бітним блочним шифром для забезпечення ав-

© О. О. Кузнецов, Є. П. Колованова, Д. В. Іваненко, О. А. Винокурова, 2014

тентичності та, вибірково, конфіденційності інформації. У поєднанні з алгоритмом шифрування AES [8] у вигляді спеціальної функції AES-GCM цей режим був використаний для заміни алгоритму HMAC [12] в популярних криптографічних протоколах, таких як SSH [13], IPSec [14] і TLS [15].

Враховуючи певні особливості режиму GCM&GMAC слід відмітити його надзвичайно високу швидкодію [17–22]. Оскільки цей режим має мінімальну затримку і мінімальну кількість виконуваних операцій він ідеально підходить для захисту пакетів даних, зокрема в різних телекомунікаційних протоколах. Так для реалізації режиму GCM&GMAC при обробці одного 128 бітного блоку даних потрібно виконати лише одне шифрування та одне 128-бітне множення в полі Галуа. При цьому операції блокового шифрування легко конвейєрувати та розпаралелювати. Крім того корпорація Intel додала інструкцію PCLMULQDQ, виділивши її для використання GCM & GMAC [17]. Ця інструкція призначена для швидкого множення у полі Галуа $GF(2^n)$, що значно прискорює реалізацію режиму GCM&GMAC.

Вражаючі результати продуктивності були також опубліковані для режиму GCM & GMAC на ряді інших платформ. Так у [18] отримано 10,68 циклів на байт при обчисленні імітовставки на 64-розрядних процесорах Intel. У [19] повідомляється про 3,5 циклів на байт для того ж алгоритму при використанні інструкції AES-NI і PCLMULQDQ корпорації Intel. На процесорах 3-го покоління Intel із використанням бібліотек OpenSSL і NSS у [20] досягається продуктивність 2,47 циклів на байт. За рахунок використання команд паралелізму у [21] можливо збільшення продуктивності режиму GCM & GMAC, оптимізована реалізація запропонована у [22].

Захищеність режиму GCM&GMAC було перевірено в конкретній моделі безпеки [23]. При цьому повинен використовуватися блочний шифр, властивості якого не відрізняється від властивостей випадкової підстановки. Проте безпека режиму залежить не лише від властивостей застосовуваного шифру, але і від вибору унікальних векторів ініціалізації для кожного шифрування, виробленого з тим же ключем. Для будь-якої заданої комбінації ключа і вектора ініціалізації, застосування режиму GCM&GMAC обмежується шифруванням 2^{39}–2^{56} біт відкритого тексту. NIST Special Publication 800-38D включає керівні принципи для вибору вектора ініціалізації [4].

У [24] описано як зловмисник може виконати оптимальні напади на режим GCM&GMAC, які відповідають нижній межі його безпеки.

В [25] описано наявність т. з. «слабких» ключів режиму GCM&GMAC. Однак ця робота не показує більш ефективну атаку, ніж раніше відомі.

У роботі [7] розглянуто випадок існування «слабких» ключів, які призводять до виродженої роботи функції ґешування, в наслідок чого формовані імітовставки не залежать від вихідних даних, їх колізійні властивості порушуються і не відповідають теоретичним оцінкам. Існування «слабких» ключів режиму GCM&GMAC відповідає випадку формування нульового субключа поліноміального ґешування в схемі GCM&GMAC, який виникає при наявності т. з. «фіксованих точок шифру», тобто таких ключів, які шифрують нульову послідовність саму у себе. Отримані у [7] теоретичні оцінки показали, що ймовірність такої події не залежить від довжини ключа, вона визначається лише довжиною блоків даних, які обробляє шифр. Відповідна емпірична оцінка числа колізій передбачає застосування зменшених моделей шифрів, зокрема, моделювання режиму вибіркового гамування із прискореним виробленням імітовставки.

Метою дослідження в цій роботі є розробка зменшеної моделі режиму вибіркового гамування із прискореним виробленням імітовставки шляхом масштабування кожного з етапів перетворень. Необхідно побудувати таку зменшену модель режиму GCM&GMAC, в якій на кожному з етапів перетворень зберігалася їх алгебраїчна структура із відповідною відтворюваністю результатів. Це надасть змогу провести емпіричні дослідження колізійних властивостей формованих імітовставок із прогнозуванням рівня криптографічного стійкості повної версії шифру, експериментально перевірити отримані у [7] теоретичні оцінки числа колізій.

3. Аналіз режиму вибіркового гамування із прискореним виробленням імітовставки відповідно до NIST SP 800-38D

Розглянемо основні перетворення режиму GCM&GMAC, зокрема проаналізуємо всі застосовувані криптографічні примітиви на можливість їхнього масштабування для проведення експериментальних досліджень колізійних властивостей формованих імітовстовок.

Для забезпечення конфіденційності відкритого тексту P застосовується функція $GCTR_K$ – деяка варіація режиму гамування [1–4, 16–22], де перший блок лічильника для зашифрування відкритого тексту генерується шляхом збільшення (inc_{32}) блоку лічильника J_0, сформованого з вектору ініціалізації IV. Для забезпечення цілісності застосовується інший механізм, який засновано на функції ґешування $GHASH_H$. Функцію $GHASH_H$ побудовано із використанням поліноміальної схеми і реалізовано через множення на фіксований параметр, який названо субключем ґешування H, з операціями в двійковому полі Галуа. Функцію ґешування використовують для стискання додаткових автентифікованих даних A (Additional Authenticated Data – AAD) та шифртекст у C в єдиний блок, який далі проходить зашифрування для створення тегу автентифікації T (імітовставки).

Зашифрування та формування тегу автентифікації (імітовсавки) виконується функцією $GCM-AE_K(IV, P, A)= (C,T)$, структурну схему обчислення якої наведено на рис. 1 [4], де позначення 0^s визначає рядок довжини s, який складається з бітів '0'; len(X) – бітова довжина рядка X; функція $[x]_s$ повертає бінарне представлення x як рядка бітової довжини s; функція $MSB_s(X)$ повертає s найбільш значущих бітів X; CIPH – затверджений 128-бітний блоковий симетричний шифр.

Розглянемо алгоритм зашифрування та отримання тегу (коду) автентифікації (імітовставки) [4].

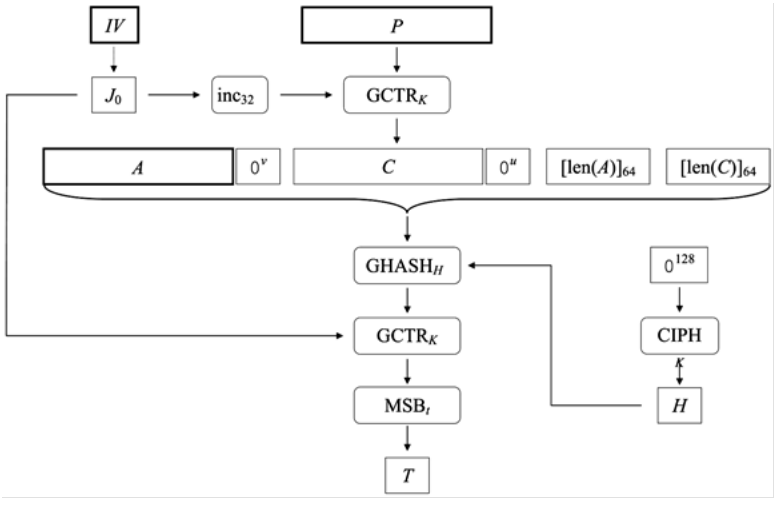

Рис. 1. Структурна схема роботи алгоритму зашифрування та отримання тегу автентифікації

Алгоритм GCM-AEK (IV, P, A)

Передумови:

затверджений блоковий шифр CIPH з 128-бітним розміром блоку;

ключ K;

визначення дозволених довжин вхідних-вихідних даних;

дозволена довжина тегу t, зв'язаного з ключем.

Вихідні дані:

вектор ініціалізації IV (дозволеної довжини);

відкритий текст P (дозволеної довжини);

додаткові автентифіковані дані A (дозволеної довжини).

Вихідні дані:

шифртекст C; тег автентифікації T.

Кроки:

Формування субключа гешування для функції $GHASH_H$

1) $$H = CIPH_K\left(0^{128}\right). \quad (1)$$

2) Визначити блок J_0 *з вектору ініціалізації IV:*

Якщо $\text{len}(IV) = 96$, *тоді*

$$J_0 = IV \| 0^{31} \| 1. \quad (2)$$

Якщо $\text{len}(IV) \neq 96$, *тоді*

$$s = 128\lceil \text{len}(IV)/128 \rceil - \text{len}(IV), \quad (3)$$

та

$$J_0 = GHASH_H(IV \| 0^{s+64} \| [\text{len}(IV)]_{64}). \quad (4)$$

3) Отримання шифртексту C

$$C = GCTR_K\left(inc_{32}\left(J_0\right), P\right). \quad (5)$$

4) Нехай

$$u = 128\lceil \text{len}(C)/128 \rceil - \text{len}(C),$$

$$v = 128\lceil \text{len}(A)/128 \rceil - \text{len}(A). \quad (6)$$

5) Визначити блок S *наступним чином:*

$$S = GHASH_H \times$$

$$\times\left(A \| 0^v \| C \| 0^u \| \left[\text{len}(A)\right]_{64} \| \left[\text{len}(C)\right]_{64}\right). \quad (7)$$

6) Нехай

$$T = MSB_t(GCTR_K(J_0, S)). \quad (8)$$

7) Повернути (C, T).

Зворотне перетворення реалізується функцією $GCM-AE_K(IV, P, A) = (C,T)$ та полягає в перевірці справжності шифртексту C із додатковими автентифікованими даними *A*. Структурну схему обчислення функції $GCM-AE_K(IV, P, A) = (C,T)$ наведеною на рис. 2 [4]. При підтвердженні справжності (знов обчислений тег автентифікації T' дорівнює отриманому T) виконується розшифрування шифртексту C та формується відкритий текст P.

Як видно з рис. 1, 2 основними перетвореннями нового режиму Galois/Counter Mode and GMAC є гешування даних із використанням функції $GHASH_H$ та зашифрування/розшифрування функцією $GCTR_K$. Розглянемо їх більш детально.

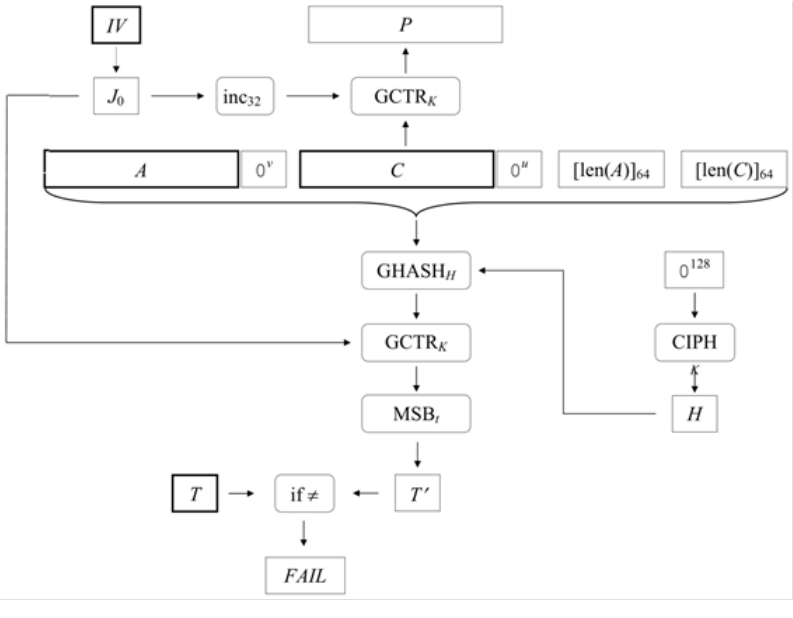

Рис. 2. Структурна схема роботи алгоритму розшифрування

3. 1. Шифрування за допомогою функції $GCTR_K$ з використанням блокового симетричного шифру AES

На рис. 3 зображено структурну схему обчислення функції $GCTR_K$ для реалізації зашифрування/розшифрування, де *ICB* – початковий блок лічильника; CB_i – і-ий блок лічильника; inc – функція інкре-

ментації. Зашифрування відбувається за допомогою функції $CIPH_K$ - блокового симетричного шифру AES зі 128-бітним розміром блоку з використанням ключа шифрування K довжиною 128 бітів.

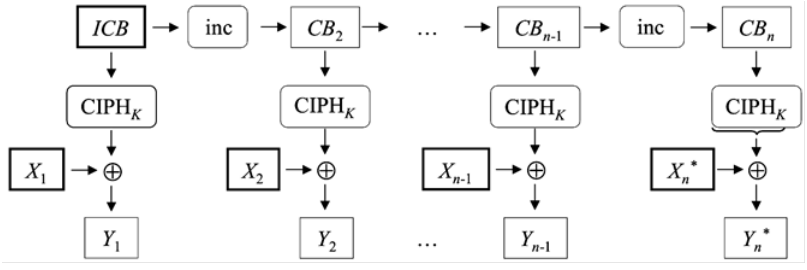

Рис. 3. Структурна схема обчислення функції шифрування

Розглянемо алгоритм зашифрування повідомлення за допомогою функції $GCTR_K$ [4].

Алгоритм $GCTR_K$ (ICB, X)

Передумови:
затверджений блоковий симетричний шифр CIPH (у даному випадку це блоковий симетричний шифр AES) зі 128-бітним розміром блоку;
ключ K довжиною 128 бітів.
Вхідні дані:
початковий блок лічильника ICB;
бітовий рядок X, довільної довжини.
Вихідні дані:
бітовий рядок Y бітової довжини len(X) – довжини рядка X.
Кроки:
1) Якщо X є пустим рядком, тоді повернути пустий рядок як Y.
2) Визначення кількості блоків довжиною 128 бітів

$$n = \lceil \mathrm{len}(X)/128 \rceil. \quad (9)$$

3) Нехай $X_1, X_2, \ldots, X_{n-1}, X_n^*$ *визначає унікальні послідовності бітових рядків таких, що*

$$X = X_1 \| X_2 \| \ldots \| X_{n-1} \| X_n^*, \quad (10)$$

причому $X_1, X_2, \ldots, X_{n-1}$ *закінчені блоки,* X_n^* *– «частковий» блок.*
4) Нехай $CB_1 = ICB$.
5) For i=2 до n

$$CB_i = \mathrm{inc}_{32}(CB_{i-1}). \quad (11)$$

6) For i = 1 до n–1

$$Y_i = X_i \oplus CIPH_K(CB_i). \quad (12)$$

7) $Y_n^* = X_n^* \oplus MSB_{len(X_n^*)}(CIPH_K(CB_n))$.

8) Формування зашифрованого повідомлення

$$Y = Y_1 \| Y_2 \| \ldots \| Y_{n-1} \| Y_n^*. \quad (13)$$

У якості затвердженого блокового симетричного шифру використовується шифр AES [8].

3. 2. Універсальне ґешування з використанням поліноміальної схеми Горнера

Для забезпечення цілісності інформації використовується функція ґешування $GHASH_H$ [4], структурна схема її обчислення зображена на рис. 4. Цю функцію побудовано на основі поліноміальної схеми [10] та реалізовано за допомогою множення на фіксований параметр – субключ. Її призначено для стискання шифртексту та додаткових автентифікованих даних в єдиний блок. Всі операції проводяться в двійковому полі Галуа. На вхід функції подається деяка унікальна послідовність блоків довжиною 128 бітів кожний, ґеш-значенння розраховуєтьс я з використанням схеми Горнера.

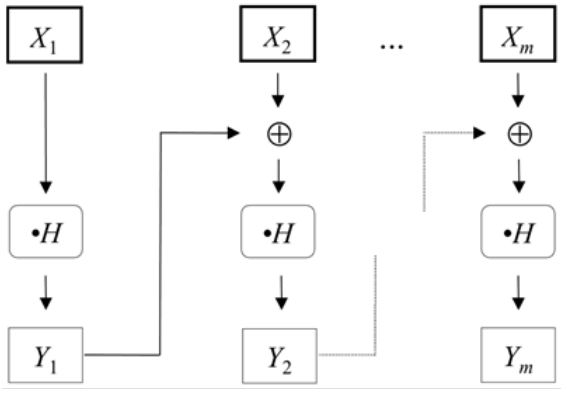

Рис. 4. Структурна схема обчислення функції ґешування

Розглянемо алгоритм роботи функції ґешування $GHASH_H$.

Алгоритм $GHASH_H$

Кроки:
1) Нехай $X_1, X_2, \ldots, X_{m-1}, X_m$ *визначає унікальну послідовність блоків, таку, що*

$$X = X_1 \| X_2 \| \ldots \| X_m. \quad (14)$$

2) Нехай Y_0 *буде «нульовим блоком»,* 0^{128}.
3) Для $i = 1, \ldots, m$, *нехай* $Y_i = (Y_{i-1} \oplus X_i) \bullet H$.
4) Повернути Y_m.

Таким чином функція $GHASH_H$ розраховує

$$Y_m = X_1 \bullet H^m \oplus X_2 \bullet H^{m-1} \oplus \ldots \oplus X_{m-1} \bullet H^2 \oplus X_m \bullet H^1, \quad (15)$$

де $X_1, X_2, \ldots, X_{m-1}, X_m$ – послідовність блоків повідомлення, H – субключ ґешування.

Функція ґешування перетворює вхідні повідомлення, довжина яких є кратною 128 (це довжина блоку), та стискає їх до загального розміру (довжини) 128 бітів.

Необхідно зазначити, що субключ ґешування H отримано зашифруванням “нульового” вектору довжиною 128 бітів за допомогою блокового симетричного шифру AES на деякому ключі K. Детально процес шифрування розглянуто у [8].

Використана функція ґешування за поліноміальною схемою Горнера належить до класу універсальних ґеш-функцій [9, 10].

Проведений аналіз основних перетворень режиму вибіркового гамування із прискореним виробленням імітовставки показав, що структура GCM&GMAC складається із декількох шарів: шифрування у режимі лічильника вибіркових даних, поліноміальне ґешування отриманих шифртекстів, шифрування у режимі лічильника формованого ґеш-значення. Тобто режим GCM&GMAC має прозору алгебраїчну структуру, яка може бути масштабована шляхом зменшення розрядності відповідних векторів відкритого тексту, ключа, шифртексту та ґеш-значення.

Таким чином, моделювання режиму вибіркового гамування із прискореним виробленням імітовставки через розробку зменшеної (міні) версії основних перетворень надасть змогу провести експериментальні дослідження колізійних властивостей формованих імітовставок із прогнозуванням рівня криптографічного стійкості повної версії шифру, експериментально перевірити отримані у [7] теоретичні оцінки числа колізій та надати обґрунтовані рекомендації по застосуванню режиму GCM&GMAC в Україні.

4. Розробка зменшеної (міні) моделі режиму вибіркового гамування із прискореним виробленням імітовставки відповідно до NIST SP 800-38D

Розглянемо зменшену модель формування кодів автентифікації повідомлень mini-GMAC, яка при збереженні математичної структури основних перетворень за рахунок зменшення ключового простору та простору автентифікаторів дозволяє оцінити кількість виникаючих колізій.

Схема формування кодів автентифікації повідомлень GMAC використовує в своїй структурі декілька етапів перетворень (в тому числі блоковий симетричний шифр AES). Зменшена модель GMAC має включати відповідні етапи перетворень зі збереженням їх алгебраїчної структури при виконанні масштабування до міні-версії.

Загальна схема формування кодів GMAC складається з таких основних етапів:

– отримання субключа ґешування H за допомогою блокового симетричного шифру mini-AES;

– формування блоку J_0 з вектору ініціалізації IV;

– формування шифртексту C з використанням функції $GCTR_K$;

– універсальне ґешування шифртексту C та додаткових автентифікованих даних із застосуванням функції $GHASH_H$ для формування єдиного кінцевого блоку (ґеш-коду);

– криптографічне перетворення з використанням блокового симетричного шифру AES для зашифрування кінцевого блоку із застосуванням функції $GCTR_K$ та блоку J_0;

– заключне перетворення для формування тегу автентифікації повідомлень шляхом обрізання результату шифрування до визначеної довжини t.

Розглянемо кожен етап схеми формування кодів автентифікації повідомлень mini-GMAC. Побудуємо міні-версію схеми формування кодів автентифікації повідомлень без зміни структури алгебраїчних перетворень звичайним зменшенням розмірності блоків оброблюваних даних у вісім разів. Таким чином, довжина кожного блоку складатиме 16 бітів, тобто всі розрахунки ведуться в кінцевому полі Галуа $GF(2^{16})$. Відповідно і довжина ґеш-коду зменшеної моделі буде дорівнювати 16 бітам.

4. 1. Отримання субключа ґешування H за допомогою блокового симетричного шифру mini-AES

Згідно з рисунком 1 на першому етапі для функції $GHASH_H$ повної версії режиму вибіркового гамування із прискореним виробленням імітовставки генерується субключ ґешування H із застосуванням блокового шифру AES на вхідному ключі K до "нульового" блоку (вираз (1)).

Для отримання субключа ґешування H в зменшеній моделі mini-GMAC будемо застосовувати схему mini-AES, яка докладно описана в [11]. Тобто будемо застосовувати довжину ключа K та довжину "нульового" блоку у 16 бітів.

Розглянемо формування субключа H на прикладі. Нехай $K = 1111\ 0100\ 0000\ 1110$. Застосуємо відповідно до [11] блоковий шифр mini-AES на вхідному ключі K до "нульового" блоку $O = 0000\ 0000\ 0000\ 0000$. Отримуємо субключ ґешування $H = 0110\ 1101\ 1001\ 1000$.

4. 2. Формування блоку J_0 з вектору ініціалізації IV

На другому етапі роботи алгоритму формується блок J_0 з вектору ініціалізації IV. Застосуємо той самий коефіцієнт масштабування, що і в п. 3. 1, зокрема, коли довжина вектору ініціалізації IV дорівнює 12 бітам, тоді для створення блоку J_0 вектор ініціалізації IV доповнюється конкатенацією рядком $0^3\|1$ згідно з (2). В іншому випадку вектор ініціалізації IV заповнюється мінімальним числом "нульових" бітів, таким чином, щоб довжина результуючого рядка була кратною 16 бітам (розміру блоку), цей рядок в свою чергу додається до 8 додаткових "нульових" бітів, після яких йде 8-бітне представлення довжини вектору ініціалізації IV, та функція $GHASH_H$ застосовується до результуючого рядка для створення блоку J_0. Тобто згідно з (4)

$$J_0 = \mathrm{GHASH}_H\left(\mathrm{IV}\|0^{s+8}\|\left[\mathrm{len}(\mathrm{IV})\right]_8\right),$$

де відповідно до (3)

$$s = 16\cdot\left\lceil \mathrm{len}(\mathrm{IV})/16\right\rceil - \mathrm{len}(\mathrm{IV}).$$

Розглянемо формування блоку J_0 з вектору ініціалізації IV на прикладі.

Спочатку розглянемо випадок, коли довжина вектору ініціалізації IV дорівнює 12 бітам, тобто

$$\mathrm{len}(\mathrm{IV}) = 12,\ \mathrm{IV} = 0100\ 0100\ 0100,$$

тоді

$$J_0 = \mathrm{IV}\|0^3\|1 = 0100\ 0100\ 0100\|000\|1 = 0100\ 0100\ 0100\ 0001.$$

Розглянемо випадки, коли $\mathrm{len}(\mathrm{IV}) \neq 12$.

1) Нехай

$$\mathrm{len}(\mathrm{IV}) = 8,\ \mathrm{IV} = 0100\ 0100.$$

Тоді

$$s = 16\cdot\lceil 8/16\rceil - 8 = 8,\ \left[\mathrm{len}(\mathrm{IV})\right]_8 = [8]_8 = 0000\ 1000,$$

$$J_0 = \mathrm{GHASH}_{\mathrm{H}}\left(\mathrm{IV}\|0^{8+8}\|[8]_8\right) =$$

$$= \mathrm{GHASH}_{\mathrm{H}}\left(01000100\|0000000000000000\|00001000\right),$$

тобто маємо

$$J_0 = \mathrm{GHASH}_{\mathrm{H}}(0100\ 0100\ 0000\ 0000\ 0000\ 0000\ 0000\ 1000).$$

Застосувавши функцію ґешування $\mathrm{GHASH}_{\mathrm{H}}$ (алгоритм роботи якої буде докладно розглянутий пізніше), отримаємо шуканий блок J_0

$$J_0 = 0101\ 1110\ 1111\ 0000.$$

2) Нехай

$$\mathrm{len}(\mathrm{IV}) = 16,\ \mathrm{IV} = 0100\ 0100\ 0100\ 0100,$$

тоді

$$s = 16\cdot\lceil 16/16\rceil - 16 = 16\cdot 1 - 16 = 0,$$

$$\left[\mathrm{len}(\mathrm{IV})\right]_8 = [16]_8 = 0001\ 0000,$$

$$J_0 = \mathrm{GHASH}_{\mathrm{H}}\left(\mathrm{IV}\|0^{0+8}\|[16]_8\right) =$$

$$= \mathrm{GHASH}_{\mathrm{H}}\left(0100010001000100\|00000000\|00010000\right),$$

тобто маємо

$$J_0 = \mathrm{GHASH}_{\mathrm{H}}(0100\ 0100\ 0100\ 0100\ 0000\ 0000\ 0001\ 0000).$$

Застосувавши функцію ґешування $\mathrm{GHASH}_{\mathrm{H}}$, отримаємо шуканий блок J_0

$$J_0 = 0010\ 0001\ 1010\ 0000.$$

3) Нехай

$$\mathrm{len}(\mathrm{IV}) = 20,\ \mathrm{IV} = 0100\ 0100\ 0100\ 0100\ 0100,$$

тоді

$$s = 16\cdot\lceil 20/16\rceil - 20 = 16\cdot 2 - 20 = 12,$$

$$\left[\mathrm{len}(\mathrm{IV})\right]_8 = [20]_8 = 0001\ 0100,$$

$$J_0 = \mathrm{GHASH}_{\mathrm{H}}\left(\mathrm{IV}\|0^{12+8}\|[20]_8\right) =$$

$$= \mathrm{GHASH}_{\mathrm{H}}\left(01000100010001000100\|00000000000000000000\|00010100\right),$$

тобто маємо

$$J_0 = \mathrm{GHASH}_{\mathrm{H}}(0100010001000100010000000000000000000000010100).$$

Застосувавши функцію ґешування GHASHH, отримаємо блок J_0

$$J_0 = 1100\ 0000\ 1111\ 0111.$$

4. 3. Формування шифртексту C з використанням функції mini-$\mathrm{GCTR}_{\mathrm{K}}$

Наступним кроком є формування шифртексту. Для цього на вхід функції $\mathrm{GCTR}_{\mathrm{K}}$ подається відкритий текст та початковий блок лічильника, який в свою чергу формується із застосуванням функції інкрементації до блоку лічильника – вираз (5). Для збереження масштабуючого коефіцієнту до блоку J_0 застосуємо 4-бітну функцію інкрементації. Результатом роботи функції mini-$\mathrm{GCTR}_{\mathrm{K}}$ із використанням ключа K є шифртекст C.

Розглянемо роботу 4-бітної функції інкрементації до блоку J_0, сформованого з вектору ініціалізації IV, причому $\mathrm{len}(\mathrm{IV}) = 12$.

$$\mathrm{inc}_4(J_0) = \mathrm{MSB}_{\mathrm{len}(J_0)-4}(J_0)\left\|\left[\mathrm{int}\left(\mathrm{LSB}_4(J_0)\right)+1 \bmod 2^4\right]_4 =\right.$$

$$= \mathrm{MSB}_{16-4}(J_0)\left\|\left[\mathrm{int}\left(\mathrm{LSB}_4(J_0)\right)+1 \bmod 2^4\right]_4 =\right.$$

$$= \mathrm{MSB}_{12}(J_0)\left\|\left[\mathrm{int}\left(\mathrm{LSB}_4(J_0)\right)+1 \bmod 2^4\right]_4,\right.$$

де $\mathrm{MSB}_s(J_0)$ – функція, яка для заданого бітового рядка J_0 повертає s найменш значущих (тобто найправіших) бітів J_0; $\mathrm{LSB}_s(J_0)$ – функція, яка для заданого бітового рядка J_0 повертає s найбільш значущих (тобто найлівіших) бітів J_0; $\mathrm{int}(J_0)$ – функція, яка представляє бітовий рядок J_0 як ціле число; $[x]_s$ – функція, яка повертає бінарне представлення x як рядка бітової довжини s з найменш значущим бітом праворуч.

Нехай

$$J_0 = 0100\ 0100\ 0100\ 0001,$$

тоді

$$\mathrm{MSB}_{12}(J_0) = 0100\ 0100\ 0100,\ \mathrm{LSB}_4(J_0) = 0001,$$

$$\mathrm{int}\left(\mathrm{LSB}_4(J_0)\right) = \mathrm{int}(0001) = 1,$$

$$\left[\mathrm{int}\left(\mathrm{LSB}_4(J_0)\right)+1 \bmod 2^4\right]_4 =$$

$$= \left[\mathrm{int}(0001)+1 \bmod 2^4\right]_4 = \left[1+1 \bmod 2^4\right]_4 =$$

$$= [2]_4 = 0010,$$

$$\mathrm{inc}_4(J_0) = 0100\ 0100\ 0100\ 0010.$$

Розглянемо алгоритм отримання шифртексту, тобто роботу функції mini-$\mathrm{GCTR}_{\mathrm{K}}$

$$C = \mathrm{GCTR}_{\mathrm{K}}\left(\mathrm{inc}_4(J_0), P\right),$$

де P – відкритий текст,

$$P = 0011\ 0011\ 0011\ 0011\ 1100\ 1100\ 1100\ 1100;$$

K – ключ шифрування,

$$K = 1111\ 0100\ 0000\ 1110.$$

1) Визначимо кількість блоків, на які розбивається відкритий текст з урахуванням коефіцієнту масштабування та (9)

$$n = \lceil \mathrm{len}(P)/16\rceil = \lceil 32/16\rceil = 2.$$

2) Розіб'ємо відкритий текст на 2 блоки відповідно до (10)

$$P = P_1 \| P_2.$$

Тоді

$$P_1 = 0011\ 0011\ 0011\ 0011,\ P_2 = 1100\ 1100\ 1100\ 1100.$$

3) Формуємо послідовність блоків лічильника (вираз (11))

$$CB_1 = inc_4(J_0) = 0100\ 0100\ 0100\ 0010,$$

$$CB_2 = inc_4(CB_1) = 0100\ 0100\ 0100\ 0011.$$

4) Формуємо шифртекст C відповідно до (12) та (13)

$$C = C_1 \| C_2,$$

де

$$C_1 = P_1 \oplus CIPH_K(CB_1),\ C_2 = P_2 \oplus CIPH_K(CB_2),$$

$CIPH_K(CB_i)$ – застосування блокового симетричного шифру mini-AES з ключем шифрування K до відповідного блоку лічильника CB_i.

Застосувавши шифр mini-AES з ключем шифрування K до блоків лічильника CB_1 та CB_2, отримуємо такі значення

$$CIPH_K(CB_1) = 1111\ 0000\ 1111\ 1010,$$

$$CIPH_K(CB_2) = 1101\ 0000\ 1111\ 0000,$$

$$C_1 = P_1 \oplus CIPH_K(CB_1) =$$
$$= 0011001100110011 \oplus 1111000011111010 =$$
$$= 1100001111001001,$$

$$C_2 = P_2 \oplus CIPH_K(CB_2) =$$
$$= 1100110011001100 \oplus 1101000011110000 =$$
$$= 0001110000111100.$$

Тоді

$$C = C_1 \| C_2 =$$
$$= 1100001111001001 \| 0001110000111100 =$$
$$= 1100001111001001000111000011110 0.$$

4. 4. Універсальне ґешування шифртексту C та додаткових автентифікованих даних із застосуванням функції mini-GHASH$_H$

На наступному етапі до додаткових автентифікованих даних та шифртексту додається мінімальна кількість "нульових" бітів, так, щоб бітова довжина результуючих рядків була кратною розміру блока. Після цього отримані рядки конкатенуються, а результат доповнюється представленням додаткових автентифікованих даних та шифртексту. Функція $GHASH_H$ застосовується до отриманого результуючого рядка для створення єдиного кінцевого блоку S. Оскільки у зменшеній моделі застосовується масштабування, блок S побудуємо наступним чином з урахуванням коефіцієнту масштабування та виразів (6) та (7):

$$S = GHASH_H\left(A \| 0^v \| C \| 0^u \| [len(A)]_8 \| [len(C)]_8\right),$$

де

$$v = 16 \cdot \lceil len(A)/16 \rceil - len(A),$$

$$u = 16 \cdot \lceil len(C)/16 \rceil - len(C).$$

Визначимо на прикладі вхідні дані для ґешування шифртексту C та додаткових автентифікованих даних із застосуванням функції mini-$GHASH_H$.

Нехай субключ ґешування $H = 1010\ 0001\ 1000\ 1110$, додаткові автентифіковані дані $A = 0001\ 0001\ 0001$, довжина рядка додаткових автентифікованих даних $len(A) = 12$, шифртекст

$$C = 1111\ 1001\ 1111\ 1100\ 0010\ 0110\ 0000\ 0101,$$

довжина рядка шифртексту $len(C) = 32$. Тоді

$$v = 16 \cdot \lceil 12/16 \rceil - 12 = 4,\ u = 16 \cdot \lceil 32/16 \rceil - 32 = 0,$$

$$S = GHASH_H(X),$$

де

$$X = A \| 0^4 \| C \| 0^0 \| [12]_8 \| [16]_8 =$$
$$= 000100010001 \| 0000 \| 11000011110010010001110000111100 \| 00001100 \| 00010000 =$$
$$= 00010001000100001100001111001001000111000011110000011000010000.$$

Розбиваємо X на n рядків згідно з (14), де

$$n = len(X)/16 = 64/16 = 4,$$

тобто

$$X = X_1 \| X_2 \| X_3 \| X_4.$$

Тоді

$$X_1 = 0001000100010000,\ X_2 = 1100001111001001,$$

$$X_3 = 0001110000111100,\ X_4 = 0000110000010000.$$

Згідно з рис. 4 та алгоритмом ґешування за допомогою функції $GHASH_H$ значення кінцевого блоку S обчислюється за поліноміальною схемою. Визначимо зменшену її модель відповідно до (15) наступним чином:

$$S = X_1 \cdot H^4 \oplus X_2 \cdot H^3 \oplus X_3 \cdot H^2 \oplus X_4 \cdot H^1.$$

Виконуючи розрахунки у полі $GF(2^{16})$ отримаємо такий результат:

$$S = 0101\ 1110\ 0010\ 0011.$$

4. 5. Криптографічне перетворення з використанням блокового симетричного шифру mini-AES для зашифрування кінцевого блоку із застосуванням функції mini-GCTR$_K$ та блоку J_0

На передостанньому етапі отриманий стиснутий за допомогою функції mini-$GHASH_H$ блок S зашифро-

вується із застосуванням функції mini-$GCTR_K$ з блоком J_0.

Застосуємо функцію mini-$GCTR_K$ з блоком J_0 (алгоритм її роботи детально описаний раніше), де у якості відкритого тексту виступає стиснутий блок S, а результатом є зашифроване повідомлення Y.

Нехай, як і у попередніх прикладах, вхідними даними є:

$$J_0 = 0100010001000001,\ K = 1111010000001110,$$

$$S = 0101\ 1110\ 00100011.$$

Тоді згідно з (9)–(13) маємо:

1) $n = \lceil \mathrm{len}(S)/16 \rceil = \lceil 16/16 \rceil = 1,$

2) $S = S_1 = 0101\ 1110\ 0010\ 0011,$

3) $CB_1 = J_0 = 0100\ 0100\ 0100\ 0001,$

4) $Y = Y_1$, де $Y_1 = S_1 \oplus CIPH_K(CB_1)$.

Застосувавши шифр mini-AES з ключем шифрування K до блоку лічильника CB_1, отримуємо наступний результат

$$CIPH_K(CB_1) = 0110\ 0000\ 1111\ 1001,$$

$$Y_1 = S_1 \oplus CIPH_K(CB_1) =$$

$$= 0101111000100011 \oplus 0110000011111001 =$$

$$= 0011111011011010.$$

Тобто $Y = 0011\ 1110\ 1101\ 1010$.

4. 6. Заключне перетворення для формування тегу автентифікації повідомлень шляхом обрізання результату зашифрування до визначеної довжини

Для формування тегу автентифікації T зашифрований рядок обрізається до визначеної довжини t: $T = MSB_t(GCTR_K(J_0, S))$. Для виконання цієї операції у зменшеній моделі режиму будемо застосовувати описані раніше зменшені версії криптоперетворень. Наприклад, сформуємо тег автентифікації при $t = 13$, а в якості $GCTR_K(J_0, S)$ виступає значення $Y = 0011\ 1110\ 1101\ 1010$. Тоді $T = MSB_{13}(Y) = 0011\ 1110\ 1101\ 1$ – у відповідності з виразом (8).

Таким чином, розроблена зменшена модель режиму GCM&GMAC зберігає алгебраїчну структуру всіх основних криптоперетворень, що дозволяє за рахунок їхнього масштабування проводити експериментальні дослідження колізійних властивостей формованих імітовставок з подальшим прогнозуванням рівня криптографічного стійкості повної версії шифру.

5. Висновки

В даній роботі проведено аналіз алгоритму роботи режиму вибіркового гамування із прискореним виробленням імітовставки відповідно до NIST SP 800-38D. Проведення експериментальних досліджень повної версії цього режиму для оцінки ймовірності виникнення колізій при формування кодів автентифікації повідомлень є неможливим з причини великої розмірності оброблюваних блоків даних. Рішення цієї задачі знайдено у розробці зменшеної моделі вибіркового гамування із прискореним виробленням імітовставки та представлено у цій роботі.

Як видно із останнього прикладу розроблена зменшена модель режиму GCM&GMAC дозволяє формувати імітовставку визначеної довжини, при цьому зберігається вся алгебраїчна структура застосовуваних перетворень, тобто запропонована модель за видами перетворень виповідає рекомендаціям стандарту NIST SP 800-38D і відрізняється лише розрядністю відповідних блоків, яка визначається встановленим коефіцієнтом масштабування. Це дозволяє стверджувати про адекватність розробленої моделі об'єкту досліджень, наведені приклади дозволяють переконатися у відтворюваності результатів моделювання.

Таким чином, масштабування застосованих перетворень на відповідних етапах схеми формування кодів автентифікації повідомлень дозволяє побудувати зменшену модель GCM&GMAC, експериментально дослідити колізійні властивості сформованих кодів. Коефіцієнт масштабування при розробці міні-моделі GCM&GMAC обрано таким чином, щоб довжина сформованих ґеш-кодів S, псевдовипадкової послідовності Y та кодів автентифікації повідомлень T дорівнювала довжині блоку міні-версії блокового симетричного шифру AES [11], тобто 16 бітам. Вибір такого коефіцієнту масштабування дозволяє зберегти алгебраїчну структуру основних перетворень алгоритму GCM&GMAC, в тому числі і схему алгоритму AES, що в нього входить. Також це дає можливість в подальшому провести експериментальні дослідження з використанням методів статистичної перевірки припущень, розглядаючи обмежений набір елементів S, Y, T та відповідні результати з оцінки числа колізій як вибірку з генеральної сукупності. Проведення таких досліджень є перспективним напрямком подальших робіт, що спрямовані, зокрема, на обґрунтування рекомендацій та певних обмежень у застосуванні режиму вибіркового гамування із прискореним виробленням імітовставки в Україні.

Література

1. ГОСТ 28147-89. Системы обработки информации. Защита криптографическая. Алгоритм криптографического преобразования [Текст]. – Введ. 1990-07-01. - М.: Изд-во стандартов, 1989. – 28 с.
2. ГОСТ Р ИСО/МЭК 10116-93. Информационная технология. Режимы работы для алгоритма n-разрядного блочного шифрования [Текст]. – Введ. 1993-12-28. - М.: Изд-во стандартов, 1994. – 20 с.
3. ISO/IEC 10116. Information technology – Security techniques – Modes of operation for an n-bit block cipher [Electronic resource]. – 2006. – Available at: \www/URL: http://www.iso.org.

4. Dworkin, M. NIST Special Publication 800-38. Block Cipher Modes [Electronic resource] / M. Dworkin. – 2007. – Available at: \www/URL: http://csrc.nist.gov.
5. ДСТУ. Інформаційні технології. Криптографічний захист інформації. Алгоритм симетричного блокового перетворення [Текст]: Проект стандарту. – 2-га (остаточна) редакція. – Київ: Держспоживстандарт України, 2014. - 238 с.
6. Розробка нового блокового симетричного шифру [Текст]: звіт за перший етап НДР «Алгоритм» (проміжний) / кер. І. Д. Горбенко; АТ «ІІТ». – Харків, 2014. – Том 4. – 304 с.
7. Кузнецов, О. О. Аналіз колізійних властивостей режиму вироблення імітовставок із вибірковим гамуванням [Текст] / О. О. Кузнецов, Д. В. Іваненко, Є. П. Колованова // Вісник Харківського національного університету ім. В. Н. Каразіна. Серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління». – 2014. - № 1097, Т. 23. – С. 55-71.
8. National Institute of Standards and Technology, FIPS 197 [Electronic resource]: Advanced Encryption Standard. – 2001. – Available at: \www/URL: http://www.nist.gov/aes.
9. Stinson, D. R. Universal hashing and authentication codes [Text] / D. R. Stinson // Designs, Codes and Cryptography. – 1994. – Vol. 4, № 3. – P. 369–380. doi:10.1007/bf01388651.
10. Polynomial hashing [Text] : 4,588,985 United States Patent: H 03 M 7/00, field of search 340/347 DD / Carter J. L., Wegman M. N.; International Business Machines Corporation, Armonk, N.Y. – filed Dec. 30, 1983 – May 13, 1986.
11. Phan, R.C.-W. Mini Advanced Encryption Standard (Mini-AES): A testbed for Cryptanalysis Students [Text] / Raphael Chung-Wei Phan // Cryptologia. – 2002. – Vol. 26, № 4. – P. 283–306. doi:10.1080/0161-110291890948.
12. Bellare, M. Keying hash functions for message authentication [Text] / M. Bellare, R. Canetti, H. Krawczyk // CRYPTO '96 Proceedings of the 16th Annual International Cryptology Conference on Advances in Cryptology. – 1996. - Vol. 1109. - P. 1–15. doi:10.1007/3-540-68697-5_1.
13. Igoe, K. AES Galois counter mode for the secure shell transport layer protocol [Electronic resource] / K. Igoe, J. Solinas // IETF Request for Comments 5647. – 2009. – Available at: \www/URL: http://tools.ietf.org/html/rfc5647.
14. Law, L. Suite B cryptographic suites for IPsec [Electronic resource] / L. Law, J. Solinas // IETF Request for Comments 4869. - 2007. – Available at: \www/URL: http://tools.ietf.org/html/rfc6379.
15. Salter, M. Suite B profile for transport layer security (TLS) [Electronic resource] / M. Salter, E. Rescorla, R. Housley // IETF Request for Comments 5430. - 2009. – Available at: \www/URL: http://tools.ietf.org/html/rfc5430.
16. Lemsitzer, S. Multi-gigabit GCM-AES Architecture Optimized for FPGAs [Text] / S. Lemsitzer, J. Wolkerstorfer, N. Felber, M. Braendli // Proceedings of the 9th international workshop on Cryptographic Hardware and Embedded Systems, CHES '07. – 2007. - Vol. 4727. - P. 227-238. doi:10.1007/978-3-540-74735-2_16.
17. McGrew, D. A. The Galois/Counter Mode of Operation (GCM) [Text] / D. A. McGrew, J. Viega. - 2013. - 41 p.
18. Kasper, E. Faster and Timing-Attack Resistant AES-GCM [Text] / E. Kasper, P. Schwabe // Cryptographic Hardware and Embedded Systems - CHES 2009, Lecture Notes in Computer Science. – 2009. - Vol. 5747. - P. 1-17. doi:10.1007/978-3-642-04138-9_1.
19. Misdetection of MIPS endianness & How to get fast AES calls? [Electronic resource]. – 2010. – Available at: \www/URL: http://groups.google.com/group/ cryptopp-users/msg/a688203c2314ef08.
20. Gueron, Sh. AES-GCM for Efficient Authenticated Encryption – Ending the Reign of HMAC-SHA-1? [Text] / Shay Gueron // Workshop on Real-World Cryptography. – Stanford University Jan. 9-11, 2013. – 32 p.
21. Gopal, V. Fast Cryptographic Computation on Intel Architecture Via Function Stitching [Electronic resource] / V. Gopal, W. Feghali, J. Guilford, E. Ozturk, G. Wolrich, M. Dixon, M. Locktyukhin, M. Perminov; Intel Corp. – 2010. – Available at: \www/URL: http://download.intel.com/design/intarch/PAPERS/ 323686.pdf.
22. Manley, R. A Program Generator for Intel AES-NI Instructions [Text] / R. Manley, D. Gregg // Progress in Cryptology - INDOCRYPT 2010, Lecture Notes in Computer Science. – 2010. - Vol. 6498. - P. 311-327. doi:10.1007/978-3-642-17401-8_22.
23. McGrew, D. A. The Security and Performance of the Galois/counter mode (GCM) of Operation [Text] / D. A. McGrew, J. Viega // Proceedings of INDOCRYPT 2004, Lecture Notes in Computer Science. – 2004. - Vol. 3348. - P. 343-355. doi:10.1007/978-3-540-30556-9_27.
24. Ferguson, N. Authentication Weaknesses in GCM [Electronic resource] / N. Ferguson // Microsoft Corp. – 2005. – Available at: \www/URL: http://csrc.nist.gov/groups/ST/toolkit/BCM/documents/comments/CWC-GCM/Ferguson2.pdf.
25. Saarinen, M-J. O. Cycling Attacks on GCM, GHASH and Other Polynomial MACs and Hashes [Text] / Markku-Juhani O. Saarinen // Fast Software Encryption. Lecture Notes in Computer Science. – 2012. - Vol. 7549. - P. 216-225. doi:10.1007/978-3-642-34047-5_13.