КІБЕРБЕЗПЕКА: освіта, наука, техніка
CYBERSECURITY: EDUCATION, SCIENCE, TECHNIQUE
№ 4 (20), 2023
ISSN 2663 - 4023
DOI 10.28925/2663-4023.2023.20.253271
УДК 004.056

Корнієць Віктор Анатолійович
аспірант
Інститут проблем математичних машин і систем НАН України, Київ, Україна
ORCIDID: 0000-0002-4967-8395
viktorkorniets@email.com

Черненко Роман Миколайович
аспірант кафедри інформаційної та кібернетичної безпеки
імені професора Володимира Бурячка
Київський університет імені Бориса Грінченка, Київ, Україна
ORCID 0000-0002-1439-961X
r.chernenko.asp@kubg.edu.ua

МОДИФІКАЦІЯ КРИПТОГРАФІЧНОГО АЛГОРИТМУ A5/1 ДЛЯ ЗАБЕЗПЕЧЕННЯ КОМУНІКАЦІЙ ПРИСТРОЇВ IOT

Анотація. Мережі інтернету речей є високо диверсифікованими з точки зору великої кількості пристроїв з різними характеристиками, операційних систем, алгоритмів захисту, протоколів передачі інформації. Криптографічні алгоритми не можуть однаково добре функціонувати на різних пристроях, більшість з них демонструє низькі показники швидкості шифрування та високі вимоги до пам'яті на 8-бітних пристроях класу C0. У статті розглядається модифікація криптографічного алгоритму A5/1 для застосування в мережах інтернету речей 8-бітними пристроями з обмеженими обчислювальними ресурсами. Сформовано модель загроз в якій визначено основні загрози та можливі методи для їх нейтралізації, зокрема методи криптографічного захисту. За рахунок розробленої модифікації вдалось усунути основні недоліки A5/1 у випадку застосування для захисту інформації в мережах інтернету речей, зокрема збільшити довжину ключа, підвищити імітостійкість, оптимізувати до використання на 8-бітних пристроях. Запропоновані заміни бітової обробки даних на байтову дозволили покращити криптографічні якості і підвищити зручність застосування алгоритму на пристроях з обмеженими обчислювальними ресурсами. За результатами статистичних тестів шифруюча послідовність може вважатись випадковою рівномірно розподіленою. Для застосування модифікованого алгоритму було побудовано криптографічний протокол з методами ідентифікації пристроїв та безпечного управління ключами. Запропоновані рішення були практично реалізовані та апробовані та досягнута прийнятна швидкість шифрування для багатьох застосувань на 8-бітному пристрої.

Ключові слова: Інтернет речей; криптографічний захист; A5/1; пристрої з обмеженими обчислювальними ресурсами; алгоритми шифрування; ефективність; конфіденційність; модель загроз.

ВСТУП

Постановка проблеми. Пристрої з обмеженими обчислювальними ресурсами (ПООР), що становлять значну частку IoT тісно інтегровані у всі сфери повсякденного життя, від домашніх пристроїв розумного будинку до пристроїв для медичного або військового застосування. Оскільки інформація в таких системах може бути з обмеженим доступом, необхідно забезпечити достатній рівень безпеки враховуючи обмежені ресурси електроживлення та обчислення, а також умови в яких функціонують такі пристрої. Різні алгоритми шифрування можуть бути реалізовані на ПООР, проте не

всі з них є ефективними з точки зору продуктивності та належного рівня криптозахисту, такий стан зумовлений високою диверсифікацією ПООР та їх обчислювальних можливостей. Використання неефективних алгоритмів може призвести до недостатнього рівня захисту інформаційних систем, низьку швидкість шифрування та порушення їх роботи через брак необхідних обчислювальних ресурсів для виконання корисного навантаження.

Аналіз останніх досліджень і публікацій. У роботі [1] автори оцінювали продуктивність деяких традиційних алгоритмів шифрування та 10 алгоритмів фіналістів конкурсу NIST на пристроях з обмеженими обчислювальними ресурсами. Авторами вказано на неоптимальність деяких алгоритмів на різних обчислювальних системах. Дослідження підтверджує необхідність забезпечення алгоритмів захисту на пристроях з обмеженими обчислювальними ресурсами, але з урахуванням архітектури та наявних обчислювальних можливостей. Зазначено необхідність стандартизації нових алгоритмів шифрування але не як жорстке правило, дозволяючи розробникам обирати алгоритми відповідно до конкретних можливостей платформи.

У [2] дослідники визначають найбільш вразливішою складовою IoT систему зв'язку машина-машина (M2M). Помилки в роботі мережі можуть залишитися непоміченими через обмежений контроль людини, що може призвести до загроз фізичної інфраструктури та безпеки. Автори доходять висновку, що незважаючи на наявність протоколів безпеки, кожен день з'являються нові кібератаки, що ставить під сумнів ефективність існуючих заходів, це підкреслює важливість надійних заходів безпеки в області IoT та M2M.

В [3] запропонована модифікована версія алгоритму TEA для вирішення його основних вразливостей, шляхом додавання методу генерації ключа на основі двох лінійних регістрів зсуву. Автори доходять висновку, що використання лінійних регістрів зсуву полегшує обчислення, що робить його більш пристосованим для використання в мережах IoT.

У роботі [4] автори запропонували модифікований метод шифрування заснований на DNA який використовується для спрощення та прискорення процесу шифрування для обчислення на пристроях IoT.

Проте більшість алгоритмів шифрування зокрема і конкурсантів конкурсу NIST орієнтовані на 64 бітні платформи, та демонструють досить низькі показники швидкості шифрування та високі вимоги до пам'яті на пристроях з 8-бітною архітектурою. Таким чином це зумовлює необхідність пошуку нових або модифікації існуючих методів захисту інформації, що передається пристроями з обмеженими обчислювальними ресурсами з 8-бітною архітектурою.

Мета статті. Мета статті полягає в забезпеченні захисту інформації на 8-бітних пристроях IoT шляхом модифікації алгоритму симетричного шифрування A5/1 для забезпечення достатнього рівня захищеності та високої швидкості шифрування.

РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Модель загроз безпеки інформації в мережі IoT

Аналіз загроз безпеки інформації в мережі IoT свідчить про їх певні особливості в плані реалізації та можливих наслідків для інформації, що обробляється в системі.

По-перше, слід звернути увагу, що в мережі IoT залежно від завдань, що виконуються, потенційно може циркулювати інформація з обмеженим доступом, до якої,

відповідно до законодавства можуть бути віднесені такі категорії як персональні дані, банківська та комерційна таємниця тощо. Можливо відмітити, що законодавство в сфері криптографічного захисту інформації не висуває конкретних вимог щодо реалізації відповідних механізмів і процедур.

По-друге, пристрої мережі IoT не тільки взаємодіють через потенційно небезпечне середовище, а також в багатьох випадках (системи відеоспостереження, системи сигналізації та контролю стану об'єктів критичної інфраструктури охоронні системи тощо) можуть перебувати за межами контрольованих територій та потенційно можуть бути доступні зловмисникам.

По-третє, інформація, яка передається в мережі може бути сфальсифікована або частково модифікована із зловмисною метою.

Перелічені фактори враховані в таблиці 1, що визначає модель загроз безпеки інформації в мережі IoT.

Таблиця 1

Модель загроз безпеки інформації в мережі IoT

№№	Загроза	Зміст загрози	Ймовірність
1.	Перехоплення	Отримання доступу до конфіденційної інформації	Висока
2.	Модифікація	Порушення цілісності даних що передаються	Висока
3.	Фальсифікація	Створення фіктивних даних	Середня
4.	НСД	Доступ до критичних параметрів віддалених пристроїв з метою їх спотворення або перехоплення	Висока
5.	Підміна	Підміна блоків та вузлів віддалених пристроїв з метою імітації недійсної обстановки	Середня
6.	Вірус	Ураження злоякісними кодами з боку мережі	Середня
7.	DDoS	Блокування роботи віддалених пристроїв з боку мережі	Середня
8.	Фізичний доступ	Отримання фізичного доступу до пристрою з метою порушення його функціональності, доступу до прошивки.	Середня

Зазначимо, що загрози №№ 1-3 можуть бути повністю або частково нейтралізовані методами криптографічного захисту інформації, загрози №№ 4-6 лише частково нейтралізуються криптографічними методами, останні з наведених загроз потребують застосування некриптографічних методів.

З урахуванням запропонованої моделі загроз постає питання які саме криптографічні методи блокування і нейтралізації можливо застосувати в умовах обмежених обчислювальних ресурсів віддалених пристроїв мережі IoT?

Особливості реалізації криптоалгоритму A5/1 та обґрунтування його модифікації

Переважна більшість сучасних криптографічних алгоритмів, не зважаючи на обов'язкову умову сумісності с різними програмними та апаратними платформами, що висувається під час їх проєктування [5-8], може доволі неефективно працювати на пристроях з обмеженими обчислювальними ресурсами.

Модифікація сучасних алгоритмів є дуже ризикованою справою, оскільки вона може привести до прояви досі невідомих вразливостей внаслідок недостатньо досліджених їх властивостей.

Особливу категорію алгоритмів потокового шифрування становлять ті, що побудовані на основі лінійних рекурентних схем щодо яких є багато наукових-практичних досліджень в плані оцінки їхньої криптографічної стійкості та визначення потенційних вразливостей.

Зокрема, створені для забезпечення конфіденційності інформаційного обміну в радіо інтерфейсі мережі стандарту GSM алгоритм A5/1 та його дещо «послаблена» у криптографічному розумінні версія стандарту A5/2 мають багату бібліографію [9,10] та потенціал для їх модернізації.

Слід зазначити, що згаданий вихідний криптоалгоритм A5/1 від самого початку його проєктування був націлений на забезпечення достатньо високої швидкодії (шифрування від 6400 біт/с) на пристроях з обмеженими обчислювальними ресурсами, включаючи шифрування на SIM карті.

Криптосхема алгоритму A5/1 (рис. 1) включає три регістри *R1*, *R2*, *R3* з лінійним зворотним зв'язком (РЛЗЗ) та схему керування рухом регістрів [9].

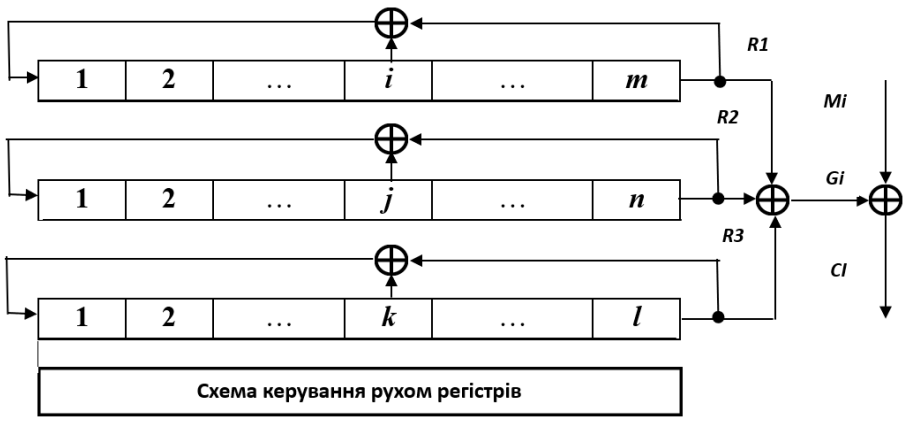

Рис. 1 – Криптосхема алгоритму A5/1

Довжина регістрів алгоритму A5/1 складає для *R1: m=19* бітів, *R2: n=22* біта, *R3: l=23* біта, їх початковий стан визначається сеансовим ключем, довжина якого становить 19+22+23=64 біта.

Біти зворотного зв'язку обрані таким чином, що характеристичний поліном кожного РЛЗЗ є примітивним, а це забезпечує максимальний період їх вихідних послідовностей (відповідно: $2^{19}-1$, $2^{22}-1$ або $2^{23}-1$). Загальний період вихідної рекурентної послідовності після їх додавання за операцією XOR в випадку рівномірного руху регістрів може становити величину $\sim 10^{19}$.

Схема керування рухом регістрів в алгоритмі A5/1 реалізує їхній нерівномірний рух від такту до такту, що спрямоване на протидію криптоаналітичним атакам. Модифікований криптоалгоритм A5/2 відрізняється від A5/1 керуванням рухом трьох РЛЗЗ за допомогою четвертого РЛЗЗ невеликої довжини з рівномірним рухом, що суттєво знижує криптографічну стійкість внаслідок можливості реалізації комбінованої атаки за схемою «перебір варіантів заповнення четвертого РЛЗЗ з можливістю розв'язання систем лінійних рівнянь.

З наведеної криптосхеми (рис. 1) не складно бачити основні можливі недоліки A5/1 (A5/2) у випадку їх застосування для захисту інформації в мережах IoT:

- невелика довжина ключу алгоритмів утворює вразливість щодо реалізації атак типу «обмін швидкості на пам'ять»;

- вузол шифрування (операція XOR) не забезпечує достатньої імітостійкості (стійкості проти підробки) в разі проведення атак на основі відомого відкритого повідомлення;
- внаслідок відносно невеликої довжини ключу існує потенційна загроза повторення шифруючої послідовності (повторення шифру) з можливістю дешифрування відповідних відкритих повідомлень;
- алгоритм не оптимізований щодо байтової структури протоколів обміну та системи команд існуючих мікроконтролерів, що ускладнює практичну програмну реалізацію операцій зсуву змісту регістрів на величину менш за 8, розрахунків значень функцій зворотного зв'язку тощо.

В той же час, можливо звернути увагу на те, що вказані алгоритмі:

- не потребують застосування команд процесора, що призводять до надмірного енергоспоживання, зокрема, операції множення та ділення;
- на відміну від блокових криптоалгоритмів не збільшують розмір зашифрованого повідомлення, у випадку коротких/не кратних розміру блоку повідомлень.

З урахуванням нормативно визначених моделей порушника доцільно визначити мінімально необхідну довжину криптографічного ключу L виходячи з потенційно можливої обчислювальної потужності його комп'ютерної техніки W (операцій за секунду), припустимого часу на реалізацію атаки τ як:

$$\tau \leq {}^{2^L}/_{W}.$$

Звідси маємо:

$$L \geq \log_2 \tau \cdot W. \quad (1)$$

Зокрема, виходячи з законодавчо визначеного терміну перегляду документів, що містять відомості кожні п'ять років та потужності сучасних суперкомп'ютерів на рівні 120-150 петафлопс з (1) отримуємо оцінку $L > 90$.

Остання нерівність та необхідність узгодження довжини ключу з форматом даних мікроконтролера дає підстави для визначення раціональної довжини ключу яка є $L = 128$.

Наступний крок – узгодження архітектури криптосхеми з системою команд 8-бітного мікроконтролера. Для цього кожну чарунку кожного регістра будемо розглядати як 8-ми бітове значення.

Загальна бітова довжина трьох РЛЗЗ має бути не менше довжини ключа:

$$L \leq 8m + 8n + 8l.$$

Остання нерівність дає оцінку $m + n + l \geq 16$. В той же час, необхідно розуміти, що загальний період T криптографічного перетворення не може перевищувати:

$$T \leq (2^m - 1) \cdot (2^n - 1) \cdot (2^l - 1) < 2^{m+n+l}. \quad (2)$$

Зауважимо, що в останній нерівності оцінка максимального періоду має місце лише за умов, що вирази в дужках попарно взаємно прості та, обов'язково мають місце попарні нерівності:

$$m \neq n,\ m \neq l,\ n \neq l. \quad (3)$$

Виходячи з ймовірності виникнення небезпечної ситуації з перекриттям шифру величина періоду перетворення не може бути менше ніж $T \geq 2^{64}$, тому з (2) отримуємо ще одну оцінку сумарної довжини регістрів в байтах:

$$m + n + l \geq 64. \quad (4)$$

На підставі (3) і (4), а також вимог щодо примітивності характеристичного полінома отримуємо варіант реалізації довжин регістрів в байтах. Зокрема, припустимим є варіант:

$$m + n + l = 19 + 22 + 23 = 64 \text{ (байти)}. \quad (5)$$

Зазначимо, що для початкового заповнення всіх трьох регістрів необхідно більше даних ніж обрана довжина ключу, яка складає загалом 16 байтів. Для узгодження цих потреб сформуємо ключовий розклад.

Позначимо: $K, \overline{K}, \widetilde{K}$ – відповідно початковий ключ, далі ключ, отриманий з початкового шляхом його циклічного зсуву на 1 байт, потім ключ, що отриманий з початкового шляхом його циклічного зсуву на 3 байти.

Під час створення початкового заповнення K завантажується в перший регістр, $\overline{K}$ завантажується в другій, а $\widetilde{K}$ – в третій регістр. Після цього не заповненими залишаються 3 байти першого регістра, 6 байт другого регістра і 7 байтів третього регістра – загалом 16 байтів, які утворюють параметр – синхромаркер S, якій використовується для перезапуску пристрою після випадкового збою.

Для забезпечення якісного початкового заповнення до початку шифрування алгоритм має відпрацювати так, щоб найдовший регістр оновився що найменш двічі. Для надійного вирівнювання пропонується забезпечити 128 тактів початкового прогону. Початковий прогін відбувається без управління рухом регістрів.

Точки зворотного зв'язку та схеми управління рухом для всіх регістрів залишаються незмінними (як в A5/1), що забезпечує необхідні якості характеристичних поліномів та управління рухом.

Відмінність полягає лише у використанні бітів в байтах управління рухом, а саме, порівняння здійснюється лише перших бітів відповідних байтів.

В процесі робочого режиму схема формує у кожному циклі роботи один байт шифруючої послідовності.

Зауважимо, що максимальний період кожної з лінійних рекурентних послідовностей в обраній схемі в кільці лишків $Z/_{2^\mu}$ згідно [11] оцінюється як:

$$T(R1) \leq (2^m - 1) \cdot 2^{\mu-1},$$
$$T(R2) \leq (2^n - 1) \cdot 2^{\mu-1},$$
$$T(R3) \leq (2^l - 1) \cdot 2^{\mu-1},$$

де m, n, l – степені відповідних характеристичних многочленів. А це означає, що у випадку вибору многочленів з урахуванням відповідних вимог [11] загальний період вихідної послідовності, якій є найменшим спільним кратним величин $T(R1)$, $T(R2)$, $T(R3)$ (без урахування схеми керування рухом регістрів), взагалі кажучи, збільшується.

В свою чергу, це свідчить про покращення криптографічних якостей модифікованої схеми.

Наступний елемент вдосконалення криптосхеми – вузол накладання шифру, який реалізований за допомогою бітової операції XOR.

В [12,13] доведено, що з точки зору забезпечення конфіденційності та імітостікості інформації найбільш ефективним вузлом накладання шифру є вузол багатоалфавітної заміни.

Формування випадкової багатоалфавітної заміни потребує суттєвих обчислювальних ресурсів, тому для модернізації вузла пропонується схема «латинський квадрат» [14,15]. Згідно з теоремою Шеннона латинські квадрати є основою ідеальних шифрів: «Ідеальні системи, в яких кількість криптограм, кількість повідомлень і кількість ключів однакові, характеризуються такими властивостями, що 1) кожне M з'єднане з кожним E рівно однією лінією, 2) усі ключі однакові. ймовірно. Таким чином, матричним зображенням системи є латинський квадрат» [14, с. 68].

В загальному випадку латинський квадрат є таблицею, в якій перший рядок є перестановкою символів, що підлягають зашифруванню, а кожний наступний рядок є результатом циклічного зсуву попереднього рядка на один символ. А саме, якщо X є підстановкою заміни степеня n, тобто

$$X = \begin{pmatrix} 1 & 2 \cdots n \\ x_1 x_2 \cdots x_n \end{pmatrix}, \tag{6}$$

тоді латинський квадрат, що породжується цією підстановкою має вигляд:

$$L(X) = \begin{pmatrix} x_1 x_2 & \cdots & x_n \\ x_2 x_3 & \cdots & x_1 \\ \cdots & \cdots & \cdots \\ x_n x_1 & \cdots & x_{n-1} \end{pmatrix}. \tag{7}$$

Рівняння зашифрування за допомогою латинського квадрату $L(X)$ має наступний вигляд:

$$C_i = L(X, M_i, G_i), \tag{8}$$

де $M_i, G_i, C_i \in \{0,1, \dots ,255\}$ – відповідно: черговий байт відкритих даних, поточне значення байту шифруючої послідовності та байт зашифрованих даних.

Суть перетворення (8) полягає у виборі в якості зашифрованого байту значення в таблиці (7) що знаходиться на перетині рядка з номером $G_i + 1$ та стовпчика $M_i + 1$.

Для нашого випадку – побайтного шифрування даних – таблиця латинського квадрату $L(X)$ матиме розмір $2^8 \times 2^8 = 65536$ (64 кбайт), що може бути дуже великим значенням для обраного ПООР.

Тому з метою уникнення необхідності збереження в пам'яті пристрою рівняння зашифрування чергового байту M_i за допомогою поточного значення G_i байту шифруючої послідовності з використанням латинського квадрату (8) можна записати у вигляді виразу:

$$C_i = X(M_i + G_i mod\ 2^8). \tag{8}$$

При цьому для збереження підстановки X потрібно лише 256 байт запам'ятовуючого пристрою.

Звернемо увагу, що використання унікальної для кожного пристрою підстановки X забезпечуватиме безпеку інших пристроїв в разі компрометації цього елементу для будь якого окремого пристрою.

Враховуючи запропоновані зміни, модифікована криптосхема алгоритму матиме вигляд як на рис. 2 та має назву A5-128.

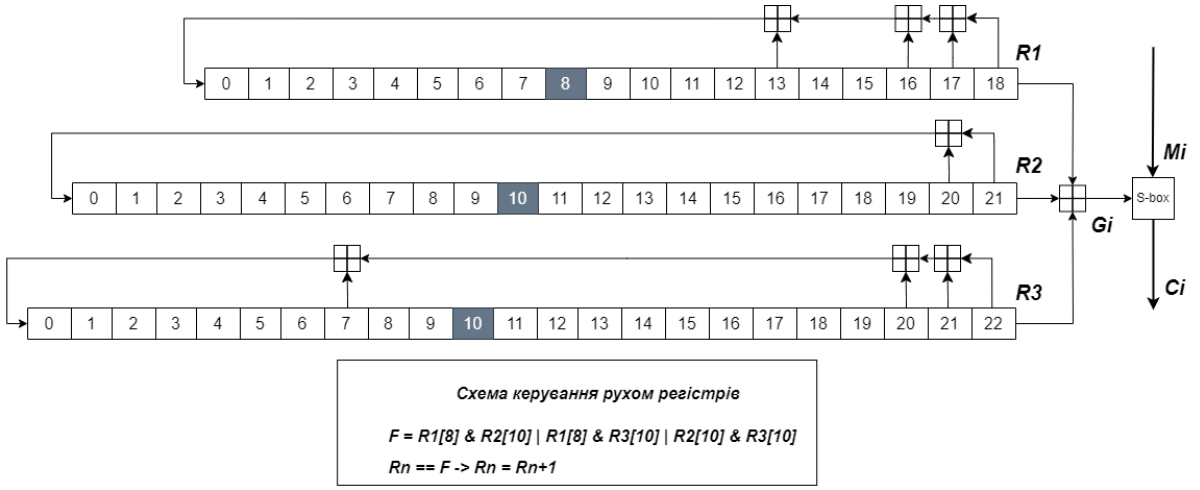

Рис. 2. Криптосхема модифікованого алгоритму A5-128

Для мінімізації кількості обчислюваних ресурсів для виконання операцій з регістрами, пропонується наступна програмна реалізація на прикладі регістру R3. Основна ідея такої реалізації полягає у відслідковуванні необхідних байтів регістру для формування нового значення та керування зсувом. Після кожного «зсуву» номер комірки зменшується на 1, необхідно тільки відслідковувати, щоб номер лежав у допустимому для регістра діапазоні, операція mod в даному випадку не використовується через те, що вона потребує більше тактів, а ніж відслідковування переповнення змінної.

```
    void shiftReg3(void) {
       reg3[reg3_22]              =              (reg3[reg3_22]              +
reg3[reg3_21]+reg3[reg3_20]+reg3[reg3_7])%256;
       if (--reg3_22 == 255) reg3_22 = 22;
       if (--reg3_21 == 255) reg3_21 = 22;
       if (--reg3_20 == 255) reg3_20 = 22;
       if (--reg3_7 == 255) reg3_7 = 22;
       if (--reg3Clock == 255) reg3Clock = 22;
    }
```

Підсумовуючи викладене, запропоновані зміни криптосхеми в частині заміни бітової обробки даних на байтову не тільки сприяють покращенню криптографічних якостей порівняно з прототипом, а й підвищують зручність застосування модифікованого алгоритму на ПООР.

За результатами статистичних досліджень шифруючих послідовностей $\{G_i\}$ на основі [16] з використанням рекомендацій [17] були отримані наступні результати.

Під час оцінки ймовірнісно-статистичних якостей модифікованої версії алгоритму A5/1 були використані рекомендації щодо інтерпретації результатів [17], де пропонується дві стратегії ухвалення рішення, щодо проходження тестів на випадковість.

Згідно *першої стратегії* необхідно визначити частку послідовностей $P1$, які пройшли перевірку, тобто $p > a$, та порівняти її з нижньою межею довірчого інтервалу $P1_{THR}$.

$$P1 = \frac{\sum_{i=1}^{1000}(P\ value(i) \geq a)}{m},\ P1_{THR} = (1-a) \pm 3\sqrt{\frac{(1-a)a}{m}} = 0.9805607$$

Якщо для одного з 15 тестів значення $P1$ виходить за ці межі, вважається, що тест не пройдено. Результати наведені в таблиці 2.

Таблиця 2

Результати статистичного тестування шифруючої послідовності згідно першої стратегії

Тест №	Назва тесту і його параметри	P1
1	Частотний монобітний	0.993
2	Частотний блочний (M=128)	0.996
3	Серій	0. 990
4	Довгих серій одиниць (M=10000)	0.992
5	Рангу випадкової {0,1}-матриці	0.989
6	Дискретного перетворення Фур'є	0.989
7…154	Відповідності аперіодичних шаблонів, що не перекриваються (M=9, 148 шаблонів)	0.990 (mean)
155	Відповідності періодичних шаблонів, що перекриваються (M=9)	0.988
156	Лінійної складності (M=500)	0.984
157	Універсальний статистичний – Маурера (L=8, Q=2356)	0.987
158	Послідовності (M=16, $\nabla\psi^2{}_m\ (obs)$)	0.996
158	Послідовності (M=16, $\nabla^2\psi^2{}_m\ (obs)$)	0.986
160	Наближеної ентропії (M=10)	0.996
161	Накопичених сум (Прямий)	0.994
162	Накопичених сум (Зворотній)	0.991
163…170	Випадкових відхилень (x = –4,…,–1, 1,…,4)	0.989
171…188	Вигляду випадкових відхилень (x = –9,…,–1, 1,…,9)	0.989

Згідно *другої стратегії* розподіл P значень для кожного тесту повинен бути рівномірно розподіленим на інтервалі [0,1].

ISSN 2663 - 4023

$$x^2 = \sum_{i=1}^{10} \frac{(Ci - m/10)^2}{m/10},\ P2 = P(x^2) = igamc(\frac{9}{2}, x^2/2).$$

Якщо отримане значення в результаті тестування $P1 < 0.0001$, то вважається, що тест не пройдений.

Для перевірки цієї стратегії значення P були розбиті на 10 підінтервалів C1-C1, з кроком 0.1. Результати наведені у таблиці 3.

Таблиця 3

Результати тестування згідно другої стратегії

Тест №	Назва тесту і його параметри	P2
1	Частотний монобітний	0.624627
2	Частотний блочний (M=128)	0.065230
3	Серій	0.518106
4	Довгих серій одиниць (M=10000)	0.117432
5	Рангу випадкової {0,1}-матриці	0.205531
6	Дискретного перетворення Фур'є	0.370262
7…154	Відповідності аперіодичних шаблонів, що не перекриваються (M=9, 148 шаблонів)	0.510992
155	Відповідності періодичних шаблонів, що перекриваються (M=9)	0.783019
156	Лінійної складності (M=500)	0.866097
157	Універсальний статистичний – Маурера (L=8, Q=2356)	0.574903
158	Послідовності (M=16, $\nabla\psi^2{}_m\ (obs)$)	0.457825
158	Послідовності (M=16, $\nabla^2\psi^2{}_m\ (obs)$)	0.984415
160	Наближеної ентропії (M=10)	0.893482
161	Накопичених сум (Прямий)	0.585209
162	Накопичених сум (Зворотній)	0.274341
163…170	Випадкових відхилень (x = –4,…,–1, 1,…,4)	0.426403
171…188	Вигляду випадкових відхилень (x = –9,…,–1, 1,…,9)	0.342937

Згідно проведеному тестуванню, модифікована версія алгоритму пройшла всі статистичні тести. Таким чином підтверджена нульова гіпотеза, і можна стверджувати що згенерована шифруюча послідовність може вважатись випадковою рівномірно розподіленою.

Отже можна стверджувати, що запропонований модифікований алгоритм за умов випадкової генерації ключа, може застосовуватись для забезпечення конфіденційності та імітостікості повідомлень що обробляються пристроями з обмеженими обчислювальними ресурсами.

Побудова захищеного протоколу для забезпечення безпеки даних в мережі IoT

Виходячи з побудованої моделі загроз для забезпечення безпеки даних в мережі IoT можливо зробити висновок, що базовими моментами для блокування загроз компрометації чутливої інформації та маніпуляцій з даними поряд з забезпеченням високих криптографічних якостей алгоритму шифрування необхідними заходами є надійна ідентифікація пристроїв, безпечне управління ключами та скорочення кількості критичних криптографічних параметрів, що зберігаються на ПООР (безумовно краще повне виключення такої можливості),

До початку сеансу захищеного обміну, необхідно згенерувати на ПООР унікальні параметри для функціонування алгоритму: ключ K, синхромаркер S, та підстановку заміни X.

Для формування підстановки заміни X можна використати швидкий алгоритм генерації підстановок багатоалфавітної заміни Складанного [12].

Згідно алгоритму Складанного для генерації підстановки степеню n: $X = \begin{pmatrix} 0 & \cdots & n-1 \\ x_0 & \cdots & x_n - 1 \end{pmatrix}$ за допомогою послідовності випадкових чисел $i_0, i_1, \dots, i_{n-1} \in Z_n$, необхідно:

1) Визначити черговий перехід: $x_k = i_m$
2) Додати перехід до множини сформованих переходів: $A = A \cup \{x_k\}$
3) Обчислити номер наступного переходу: $k = k + 1 \; mod \; n$
4) Знайти номер чергового випадкового числа $m = m + 1$, якщо $m = n$, перейти в кінець
5) Якщо $i_m \in A$ перейти до визначення наступного переходу, інакше
6) Виконати модифікацію $i_m = i_m + \delta \; mod \; n$, та перейти до перевірки визначених переходів.
7) Останньому преходу призначити $x_k = Z_n / A$

Перевагою цього алгоритму є те, що він генерує рівномірно розподілену послідовність за фіксовану кількість кроків, та є більш швидким порівняно з методом безповторного набору, що є дуже важливим на ПООР.

Для формування ключа та синхромаркера можна використовувати шум з непідключених контактів аналогово-цифрового перетворювача (АЦП). Проте необхідно враховувати лише молодші біти зчитаного значення, оскільки вони демонструють більш випадкову поведінку [18]. Така поведінка може свідчити про те, що шуми в навколишньому середовищі є невеликими, тому їх вплив можна побачити тільки на молодших бітах зчитаних значень. При цьому додавати поточне значення молодшого біта $n[i]$ тільки якщо $n[i]! = n[i-1]$.

Під час формування ключа та синхромаркера необхідно передбачити перевірку згенерованого значення за допомогою частотного монобітного тесту, задля протидії похибкам системи та потенційного зовнішнього впливу на контролер:

$$P = erfc(\frac{S_{abs}}{\sqrt{2}}), \quad (9)$$

де S_{abs} – абсолютна величина суми X_i по всій довжині послідовності, поділена на корінь квадратний з довжини, $erfc$ – комплементарна функція похибки.

Якщо отримане значення менше 0.01, послідовність необхідно відхилити та спробувати, ще раз, після певної кількості невдач, необхідно сповістити шлюз про помилку в роботі системи.

Звернемо увагу, вимога безпечного управління ключами шифрування корелюється з вимогою скорочення кількості критичних криптографічних параметрів, що зберігаються на ПООР. Дійсно, якщо для формування спільного сеансового ключу припустити використання симетричного алгоритму шифрування, то це потребує використання транспортного ключу для шифрування сеансових ключів під час передавання їх через незахищене середовище, а це породжує нову доволі складну проблему – надійний захист від несанкціонованого доступу до транспортного ключу на ПООР.

Застосування же асиметричних криптоалгоритмів хоча і призводить до певного навантаження на ПООР, але можливо відміти, наступне:

- формування сеансових ключів – це подія, яка відбувається відносно рідкісне, при цьому для відновлення сеансів зв'язку, що були розірвані внаслідок перешкод і збоїв обладнання припустимо не змінювати ключ, а генерувати новий синхромаркер;
- шифрування даних за допомогою асиметричних алгоритмів (так звана процедура – цифровий конверт) працює достатньо швидко, особливо в разі застосування перетворень, що базуються на еліптичних кривих [19,20]. При цьому зворотна процедура – розшифрування ключу має відбуватись на більш потужному обчислювальному пристрої;
- сертифікати відкритих ключів у цьому випадку сприяють реалізації процедур ідентифікації.

З урахуванням зроблених зауважень захищений протокол взаємодії ПООР і хосту в мережі IoT має включати три фази:

- реалізація процедури ідентифікації і передача ПООР сертифікату відкритого ключу хоста;
- генерація сеансового ключу на ПООР, його шифрування за допомогою асиметричного алгоритму [21] та передачу його на хост;
- потокове шифрування даних на ПООР з використанням запропонованої модифікації алгоритму A5/1. За необхідності, має відбуватись процедура відновлення зв'язку після короткочасних збоїв з використанням наявного сеансового ключу та нового випадкового синхромаркера.

Час існування сеансового ключу має бути обмеженим виходячи з умов експлуатації мережі.

Позначимо:

$Id1$ – ідентифікатор шлюзу

$Id2$ – ідентифікатор ПООР

S – синхромаркер

$C(X.509)$ – сертифікат відкритого ключа

K_e – відкритий ключ ДСТУ 9041

K_d – секретний ключ ДСТУ 9041

K_c – сеансовий ключ А5/1

$E(.,.)$ – алгоритм ДСТУ 9041

$M = (m_1, m_2, \dots, m_l)$ – послідовність відкритого повідомлення ПООР

$C = (c_1, c_2, \dots, c_l)$ – послідовність шифрованого тексту $C = A5\text{–}128(M, K_c)$

X – підстановка заміни – перший рядок латинського квадрата $L(X)$

$CRC32$ – контрольна сума

Загальний криптографічний протокол матиме вигляд (таблиця 4).

Таблиця 4

Криптографічний протокол із застосуванням алгоритму А5-128

Крок	ПООР	Крок	Шлюз
1	Отримує $C(X.509)$ та перевіряє $Id1$	1	Надсилає $C(X.509)$
2	Генерує K_c, X, S $W = K_c \,\|\|X\|\|Id1\|\|Id2\|\|CRC32$	2	Отримує V та розшифровує секретним ключем $W = E^{-1}(V, K_d)$
3	Надсилає $V = E(W, K_e)$	3	Перевіряє ідентифікатори $Id1$ та $Id2$ та контрольну суму $CRC32$
4	Шифрує повідомлення $C = A5\text{–}128(M, K_c, S)$ та надсилає $C' = C\|\|S$	4	Отримує C', ініціалізовує синхромаркер та розшифровує повідомлення.

Звернемо увагу, у випадку надсилання великої кількості коротких повідомлень, передавати кожного разу новий синхромаркер може бути недоцільно з міркувань швидкодії. Тоді зміни протоколу будуть в наступному (рис. 3):

Крок 4. ПООР передає синхромаркер S з вказівкою шлюзу зберігати поточний стан регістрів, після чого шифрує повідомлення та передає C.

Крок 4. Шлюз отримує синхромаркер та вказівку і відслідковує стан регістрів та розшифровує кожне наступне C.

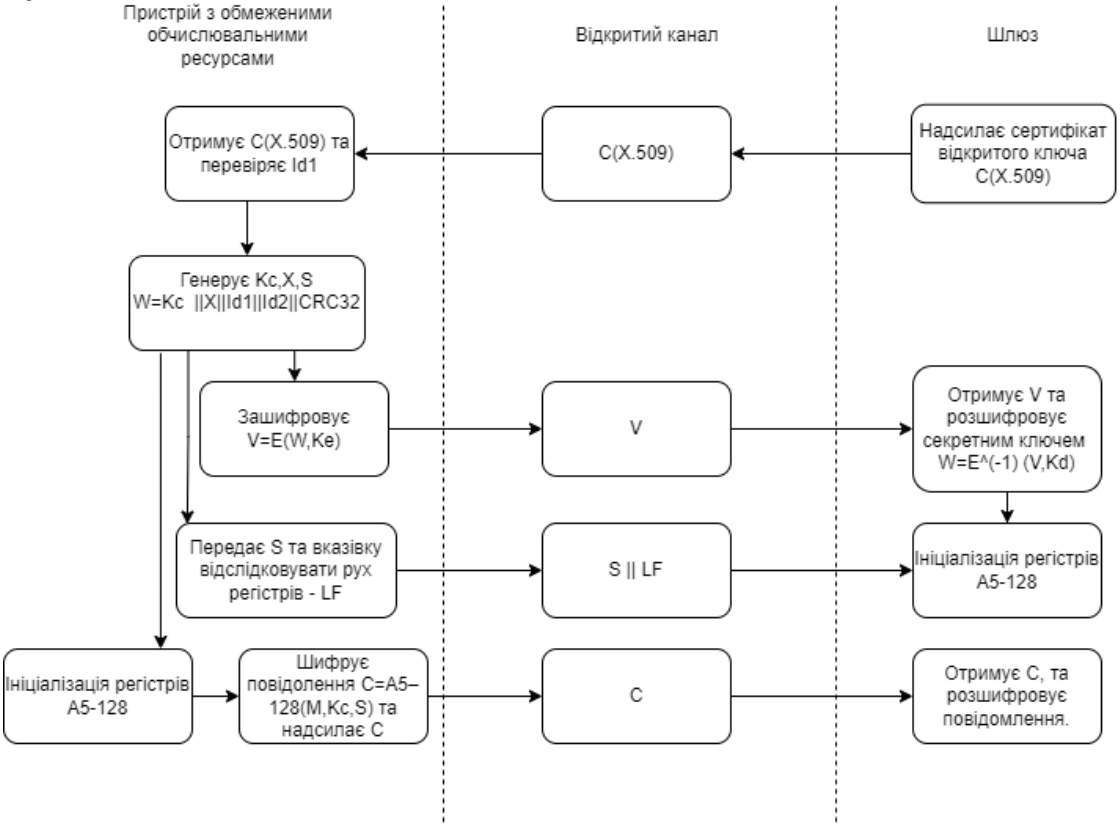

Рис. 3. Криптографічний протокол A5-128

Для впровадження алгоритму A5-128 було обрано один із бездротових протоколів, що широко застосовується в IoT застосунках Enhanced ShockBurst (ESB), що також використовується сімейством чипів радіо зв'язку nRF24L01.

ESB представляє собою протокол передачі даних на основі пакетів. Він включає автоматичне формування та синхронізацію пакетів, автоматичне підтвердження та повторну передачу пакетів. ESB дозволяє реалізацію високоефективного зв'язку з високою продуктивністю за низьку ціну з використанням мікроконтролерів низької вартості. Як зазначено в [22], мережева топологія зв'язку ESB розроблена без будь-якого вбудованого шифрування чи процесу забезпечення безпеки за замовчуванням.

Пакет ESB включає в себе поле преамбули, поле адреси, поле керування пакетом, поле корисного навантаження та поле (циклічного надлишкового коду). На рис. 4 представлено формат пакета, де старший біт розташований ліворуч.

Преамбула 1 байт	Адреса 3-5 байт	Керування пакетом 9 біт	Корисне навантаження 0 - 32 байт	CRC 1-2 байти

Рис. 4 Вигляд пакету Enhanced ShockBurst за замовчуванням

Преамбула - це послідовність бітів, яка використовується для виявлення рівнів 0 та 1 на приймачі. Преамбула складається з одного байта і може бути або 01010101, або 10101010. Якщо перший біт у адресі дорівнює 1, преамбула автоматично встановлюється в 10101010, а якщо перший біт - 0, то преамбула автоматично встановлюється в

01010101. Це робиться для забезпечення достатньої кількості переходів у преамбулі для стабілізації роботи приймача.

Адреса - це адреса для приймача. Адреса забезпечує виявлення пакетів приймачем. Поле адреси може бути налаштовано на довжину від 3 до 5 байт, адресу можна змінювати в процесі роботи.

Поле керування пакетом містить поле де вказується довжина корисного навантаження за допомогою 6 біт, поле ідентифікації пакету з 2 біт і прапор NO_ACK з 1 біта, що вказує на автоматичне підтвердження.

Корисне навантаження – це визначений користувачем вміст пакету. Він може бути від 0 до 32 байтів і передається в прямому ефірі під час завантаження (без змін) на пристрій.

CRC — це механізм виявлення помилок у пакеті. Він може складати 1 або 2 байти та обчислюється за адресою, полем керування пакетом і корисним навантаженням.

У роботах [23,24] пропонуються певні механізми забезпечення захисту інформації з використанням протоколу ESB. Недоліком цих робіт є використання вимогливих до обчислювальних ресурсів алгоритмів як AES та геш-функцій KMAC та SipHash. Як висновок [24] використання таких алгоритмів вимагає дуже багато часу на шифрування даних.

Для забезпечення захисту інформації, що передається таким протоколом, збільшення швидкості шифрування та досягнення мети використання меншої кількості обчислювальних ресурсів для забезпечення достатнього рівня захисту вирішено впровадити модифікований алгоритм шифрування.

Полям пакету яким можна задати певне значення для забезпечення захисту є адреса та корисне навантаження. Оскільки інші поля використовуються як керуючі для роботи протоколу.

Основним методом забезпечення криптозахисту даного протоколу буде шифрування корисного навантаження. Звернемо увагу, розмір корисного навантаження одного пакету обмежений розміром 32 байти, а розмір V, що передається на етапі ініціалізації очевидно буде більшим, в такому випадку перед передачею V, ПООР повідомляє ШЛЮЗ про розмір повідомлення $V: N$. Це стосується також всіх повідомлень розмір яких більше за 32 байти. Таким чином після застосування алгоритму A5-128 пакет матиме вигляд як на рис. 5.

Преамбула 1 байт	Адреса 3-5 байт	Керування пакетом 9 біт	Захищене корисне навантаження	CRC 1-2 байти

Рис. 5. Вигляд пакету Enhanced ShockBurst з шифруванням

ВИСНОВКИ ТА ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

Запропоновані в роботі рішення були практично реалізовані та апробовані. На 8-бітному контролері ATMega328P з частотою 16МГц було досягнута прийнятна для багатьох застосувань швидкість шифрування на рівні 87,7 Кбайт/сек.

Подальші дослідження доцільно зосередити на вдосконаленні процедур генерації випадкових параметрів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Rahman, M. S., Karnik, S., & Sarangerel, S. (2022). Lightweight Cryptography. MIT Course Project. Retrieved from https://courses.csail.mit.edu/6.857/2022/projects/Shahir-Rahman-Karnik-Sarangerel.pdf
2. Pradhan, D., & Tun, H. (2022). Security Challenges: M2M Communication in IoT. Journal of Electrical Engineering and Automation, 4, 187-199. https://doi.org/10.36548/jeea.2022.3.006
3. Mhaibes, H. I., Abood, M. H., & Farhan, A. (2022). Simple Lightweight Cryptographic Algorithm to Secure Embedded IoT Devices. International Journal of Interactive Mobile Technologies (iJIM), 16(20), 98–113. https://doi.org/10.3991/ijim.v16i20.34505
4. Al-Shargabi, B., & Dar Assi, A. (2023). A modified lightweight DNA-based cryptography method for Internet of Things devices. Expert Systems, 40(6), e13270. https://doi.org/10.1111/exsy.13270
5. Final report of European project number IST-1999-12324, named New European Schemes for Signatures, Integrity, and Encryption. Springer-Verlag, Berlin Heidelberg NewYork, etc. 2004, NESSIE public report D20. NESSIE Security Report. http://cryptonessie.org.
6. Daemen, J. Rijmen V. (1998) «AES Proposal: Rijndael», AES Round 1 Technical Evaluation CD1: Documentation, National Institute of Standards and Technology, Aug 1998. http://www.nist.gov/aes.
7. Горбенко І. Д., Горбенко Ю. І. (2012) Прикладна криптологія: монографія. – Харків, ХНУРЕ, Форт, 2012. – 868 с.
8. Кузнецов О. О. та ін. (2014) Обґрунтування вимог, побудування та аналіз перспективних симетричних криптоперетворень на основі блочних шифрів. URL: https://science.lpnu.ua/sites/default/files/journal-paper/2017/nov/6634/21-124-141.pdf
9. Buhantsov A.D., Sadjiid A.Yu., Ustinov A.N., Rodionov C.V. (2021) Research of speech encryption reliability in GSM mobile communication technology. Research result. Information technologies. – vol.6, №2, 2021. P. 9-17. DOI: 10.18413/2518-1092-2021-6-2-0-2
10. Xu, Y., Hao, Y., & Wang, M. (2023). Revisit two memoryless state-recovery cryptanalysis methods on A5/1. IET Information Security, 17. https://doi.org/10.1049/ise2.12120.
11. Glukhov M., Elizarov V., Nechaev A. (2003) Algebra, vol. 2, Gelios APB, 2003. 416 p. ISBN8-85338-072-2
12. Бурячок В. Л., Гулак Г.М., Складанний П. М. (2017) Швидкий алгоритм генерації підстановок багато алфавітної заміни. Захист інформації. 2017. №2. С. 173–177.
13. Гулак Г.М., Складанний П.М. (2017) Забезпечення гарантоздатності автоматизованих систем управління та передачі даних безпілотних літальних апаратів. Математичні машини і системи. 2017. № 3. С. 154–161.
14. Shannon C. (1949) Communication Theory of Secrecy Systems. Bell System Technical J. 1949. vol. 28. P. 656–715.
15. Massey J.L., Maurer U., and Wang M. Non-Expanding (1988) Key-Minimal, Robustly-Perfect, Linear and Bilinear Ciphers. Adv. Cryptology EUROCRYPT'87. Berlin; Heidelberg: Springer Verlag, 1988. P. 237–247.
16. Гулак Г., Ковальчук Л. (2001) Різні підходи до визначення випадкових послідовностей / Правове, нормативне та метрологічне забезпечення системи захисту інформації в Україні. – К., 2001. Вип. 3. С.127–133.
17. Special publication NIST SP 800-22A (2010) Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. URL: https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-22r1a.pdf
18. Kristinsson, B. (2011). Ardrand: The Arduino as a Hardware Random-Number Generator. Retrieved from https://api.semanticscholar.org/CorpusID:195592641
19. J. Hoffstein, J. Pipher, J.H. Silverman. (2014) An introduction to mathematical cryptography. – Springer. 2014. – 523 p. ISBN 978-1-4939-1711-2
20. Бессалов А.В. Еліптичні криві в формі Едвардса і криптографія: монографія. – К.: ІВЦ «Видавництво «Політехника»», 2017. –272с.
21. ДСТУ 9041:2020 Інформаційні технології. Криптографічний захист інформації. Алгоритм шифрування коротких повідомлень, заснованих на скручених еліптичних кривих Едвардса (2020). Retrieved from: http://online.budstandart.com/ua/catalog/doc-page.html?id_doc=90523
22. Kulasekara, V., Balasooriya, S., Chandran, J., & Kavalchuk, I. Novel low-power NRF24L01 based wireless network design for autonomous robots. In 2019 25th Asia-Pacific Conference on Communications (APCC), 2019 (pp. 342-346): IEEE

23. A security mechanism for Enhanced ShockBurst wireless communication protocol using nRF24L01. https://doi.org/10.21203/rs.3.rs-3777984/v1
24. Rivera, D., García, A., Martín-Ruiz, M. L., Alarcos, B., Velasco, J. R., & Oliva, A. G. (2019). Secure communications and protected data for a Internet of Things smart toy platform. IEEE internet of things Journal, 6(2), 3785-3795.

КІБЕРБЕЗПЕКА: освіта, наука, техніка
CYBERSECURITY: EDUCATION, SCIENCE, TECHNIQUE
№ 4 (20), 2023
ISSN 2663 - 4023

Viktor A. Korniiets
Postgraduate
Institute of Problems of Mathematical Machines and Systems of the National Academy of Sciences of Ukraine, Kyiv, Ukraine
ORCID ID 0000-0002-4967-8395
viktorkorniets@email.com

Chernenko M. Roman
Postgraduate of the Department of Information and Cyber Security
Borys Grinchenko Kyiv University, Kyiv, Ukraine
ORCID ID: 0000-0002-1439-961X
r.chernenko.asp@kubg.edu.ua

MODIFICATION OF THE CRYPTOGRAPHIC ALGORITHM A5/1 TO ENSURE COMMUNICATION FOR IOT DEVICES

Abstract. Internet of Things (IoT) networks exhibit high diversification due to the significant number of devices with varying characteristics, operating systems, protection algorithms, and information transmission protocols. Cryptographic algorithms, however, cannot perform equally well on different devices; most of them demonstrate low encryption speed and high memory requirements on 8-bit C0-class devices. This article explores the modification of the cryptographic algorithm A5/1 for application in IoT networks with 8-bit devices with limited computational resources. A threat model is formulated, identifying major threats and possible methods for neutralization, including cryptographic protection methods. Through the developed modification, the main drawbacks of A5/1 when applied to protect information in IoT networks have been addressed, including increasing the key length, enhancing tamper resistance, and optimizing for use on 8-bit devices. Proposed substitutions of bit data processing with byte processing have improved cryptographic qualities and made algorithm application more convenient on devices with limited computational resources. Based on statistical tests, the encrypted sequence can be considered uniformly distributed at random. For the application of the modified algorithm, a cryptographic protocol was constructed, incorporating methods for device identification and secure key management. The proposed solutions were practically implemented and tested, achieving acceptable encryption speed for many applications on an 8-bit device.

Keywords: Internet of Things; cryptographic protection; A5/1; devices with limited computational resources; encryption algorithms; efficiency; confidentiality; threat model.

REFERENCES

1. Rahman, M. S., Karnik, S., & Sarangerel, S. (2022). Lightweight Cryptography. MIT Course Project. Retrieved from https://courses.csail.mit.edu/6.857/2022/projects/Shahir-Rahman-Karnik-Sarangerel.pdf
2. Pradhan, D., & Tun, H. (2022). Security Challenges: M2M Communication in IoT. Journal of Electrical Engineering and Automation, 4, 187-199. https://doi.org/10.36548/jeea.2022.3.006
3. Mhaibes, H. I., Abood, M. H., & Farhan, A. (2022). Simple Lightweight Cryptographic Algorithm to Secure Embedded IoT Devices. International Journal of Interactive Mobile Technologies (iJIM), 16(20), 98–113. https://doi.org/10.3991/ijim.v16i20.34505
4. Al-Shargabi, B., & Dar Assi, A. (2023). A modified lightweight DNA-based cryptography method for Internet of Things devices. Expert Systems, 40(6), e13270. https://doi.org/10.1111/exsy.13270
5. Final report of European project number IST-1999-12324, named New European Schemes for Signatures, Integrity, and Encryption. Springer-Verlag, Berlin Heidelberg NewYork, etc. 2004, NESSIE public report D20. NESSIE Security Report. http://cryptonessie.org.
6. Daemen, J. Rijmen V. (1998) "AES Proposal: Rijndael," AES Round 1 Technical Evaluation CD1: Documentation, National Institute of Standards and Technology, Aug 1998. http://www.nist.gov/aes.
7. Horbenko I. D., Horbenko Yu. I. (2012) Applied Cryptology: monograph. – Kharkiv, KhNURE, Fort, 2012. – 868 p.

8. Kuznetsov O. O. et al. (2014) Substantiation of requirements, construction, and analysis of promising symmetric cryptographic transformations based on block ciphers. URL: https://science.lpnu.ua/sites/default/files/journal-paper/2017/nov/6634/21-124-141.pdf
9. Buhantsov A.D., Sadjiid A.Yu., Ustinov A.N., Rodionov C.V. (2021) Research of speech encryption reliability in GSM mobile communication technology. Research result. Information technologies. – vol.6, №2, 2021. P. 9-17. DOI: 10.18413/2518-1092-2021-6-2-0-2
10. Xu, Y., Hao, Y., & Wang, M. (2023). Revisit two memoryless state-recovery cryptanalysis methods on A5/1. IET Information Security, 17. https://doi.org/10.1049/ise2.12120.
11. Glukhov M., Elizarov V., Nechaev A. (2003) Algebra, vol. 2, Gelios APB, 2003. 416 p. ISBN8-85338-072-2
12. Buryachok V. L., Gulak G.M., Skladannyy P. M. (2017) Fast algorithm for generating substitutions of a multialphabetic substitution. Information security. 2017. №2. P. 173–177.
13. Gulak G.M., Skladannyy P.M. (2017) Ensuring reliability of automated control and data transmission systems of unmanned aerial vehicles. Mathematical machines and systems. 2017. № 3. P. 154–161.
14. Shannon C. (1949) Communication Theory of Secrecy Systems / Bell System Technical J. 1949. vol. 28. P. 656–715.
15. Massey J.L., Maurer U., and Wang M. Non-Expanding (1988) Key-Minimal, Robustly-Perfect, Linear and Bilinear Ciphers. Adv. Cryptology EUROCRYPT'87. Berlin; Heidelberg: Springer Verlag, 1988. P. 237–247.
16. Gulak G., Kovalchuk L. (2001) Different approaches to defining random sequences / Legal, regulatory, and metrological support of the information protection system in Ukraine. – Kyiv, 2001. Issue 3. P.127–133.
17. Special publication NIST SP 800-22A (2010) Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. URL: https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-22r1a.pdf
18. Kristinsson, B. (2011). Ardrand: The Arduino as a Hardware Random-Number Generator. Retrieved from https://api.semanticscholar.org/CorpusID:195592641
19. J. Hoffstein, J. Pipher, J.H. Silverman. (2014) An introduction to mathematical cryptography. – Springer. 2014. – 523 p. ISBN 978-1-4939-1711-2
20. Bessalov A.V. Elliptic curves in Edwards form and cryptography: monograph. – Kyiv: IVC "Publishing House "Polytechnika"", 2017. – 272 p.
21. DSTU 9041:2020 Information technologies. Cryptographic protection of information. Algorithm for encrypting short messages based on twisted Edwards elliptic curves (2020). Retrieved from: http://online.budstandart.com/ua/catalog/doc-page.html?id_doc=90523
22. Kulasekara, V., Balasooriya, S., Chandran, J., & Kavalchuk, I. Novel low-power NRF24L01 based wireless network design for autonomous robots. In 2019 25th Asia-Pacific Conference on Communications (APCC), 2019 (pp. 342-346): IEEE
23. A security mechanism for Enhanced ShockBurst wireless communication protocol using nRF24L01. https://doi.org/10.21203/rs.3.rs-3777984/v1
24. Rivera, D., García, A., Martín-Ruiz, M. L., Alarcos, B., Velasco, J. R., & Oliva, A. G. (2019). Secure communications and protected data for an Internet of Things smart toy platform. IEEE Internet of Things Journal, 6(2), 3785-3795.