DOI 10.31891/2307-5732-2020-289-5-115-119
УДК 004.056.5:621

Н.І. ПРАВОРСЬКА, К.Л. ГОРЯЩЕНКО, С.К. ПІДЧЕНКО
Хмельницький національний університет

РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ РЕАЛІЗАЦІЇ АЛГОРИТМУ КРИПТОГРАФІЧНОГО ПЕРЕТВОРЕННЯ В ПРОЦЕСОРНИХ СИСТЕМАХ ДЛЯ СИСТЕМ ІНТЕРНЕТУ РЕЧЕЙ

Пристрої Інтернету речей поступово розширюють зону свого застосування. Пристрої виконують задачу збору інформації, накопичення та передачу до інших пристроїв із застосуванням дротових та бездротових технологій. Показано, що для пристроїв IoT також можуть виставлятись вимоги щодо забезпечення надійності та криптостійкості даних. Для реалізації задачі шифрування можуть використовуватись як симетричні, так і асиметричні протоколи. Запропоновано використовувати асиметричний протокол на базі еліптичних кривих. В роботі виконана оцінка апаратних затрат для реалізації блоку криптографічного перетворення із застосуванням 8- та 32-бітних контролерів для ключів довжиною 32 та 64 біти та різних блоків даних. Показано практичну можливість виконання такої роботи.

Ключові слова: криптостійкість, асиметричне шифрування.

N. PRAVORSKA, K. HORIASHCHENKO, S. PIDCHENKO
Khmelnytskyi National University, Ukraine

IMPLEMENTATION OF CRYPTOGRAPHIC TRANSFORMATION ALGORITHM IN PROCESSOR SYSTEMS FOR THE INTERNET OF THINGS

Devices for Internet of Things are gradually expanding their scope. Those devices perform the task of collecting information, accumulating and transmitting to other devices with use of wired and wireless technologies. It has been shown that IoT devices can also be required to ensure the reliability and cryptographic stability of data. Both symmetric and asymmetric protocols can be used to implement the encryption task. Block encryption is a substitution based on the block alphabet, which can be mono- or polyalphabetic. Block symmetric encryption is most widely used in the transmission of information over the network, including for packet exchange systems.

It is proposed to use an asymmetric protocol based on elliptic curves. To assess the potential performance of data processing algorithms by the encryption algorithm, you need to determine the actions that use the most CPU time. Such actions include operations of bringing in a degree. To sum up, algorithms are used, the purpose of which is to perform actions on short numbers, the bit size of which is proportional to the bit size of the processor system. Processing time of alphabet values for 8-bit controllers defined. The paper estimates the hardware costs for the implementation of the cryptographic conversion unit using 8-bit and 32-bit controllers for 32 and 64 bits length keys and for different data sizes blocks. ZigBee modules often work in the mode of interval data transmission, such as transmission in the interval of minutes or even hours, in which case, it is established that the 8-bit controller will provide work on encryption and its transmission to the main module. The practical possibility of performing such work is shown. It is seen that the use of 32-bit architecture allows us to archive more than 70 times speed advantage.

Keywords: cryptosecurity, asymmetric encryption.

Вступ

Сьогодні у світі у сфері передачі інформації визначилася стабільна тенденція на посилення ролі технічних засобів захисту даних. Тенденція зовсім не випадкова: неодноразові дослідження в області безпеки даних показали, що використання технічних засобів із елементами криптографії дозволяє звести до мінімуму або виключити негативний вплив самої ненадійної ланки в системі людини. При цьому, організація розподіленої мережі контролю і передачі за допомогою технічних засобів обходиться споживачеві значно дешевше, а надійність її вища.

Історично, при створенні систем розподіленого контролю основна увага приділялася таким аспектам, як:

- автоматизація, яка дозволяє до мінімуму спростити процеси введення об'єктів під охорону, скоротити обслуговуючий персонал; суттєво скоротити кількість неправдивих тривог через втручання в роботу системи;
- контроль каналу зв'язку, що забезпечує високу достовірність передачі і виключає втрату важливої інформації;
- розробка широкої гамми об'єктових пристроїв з різними функціональними і сервісними можливостями, що дозволяють задовольнити потреби найширших верств населення.

Тому в цілях подальшого розвитку і вдосконалення систем збору та передачі інформації до нових розробок останнім часом пред'являються додаткові вимоги:

- імітостійкість і криптозахист, системи, що забезпечують стійкість до несанкціонованого «обходу» і обумовлені появою «кваліфікованих» втручань (рис. 1);
- висока інформативність системи, що забезпечує формування сигналів про втручання в спільний потік даних мережі;
- можливість інтеграції системи з оптоволоконними каналами зв'язку, обумовлена введеннням в експлуатацію підприємствами зв'язку нових цифрових технологій передачі інформації;

- уніфікація створюваних технічних засобів, тобто можливість об'єднання різних пристроїв в єдиний програмно-апаратний комплекс збору та передачі інформації.

Однією з безпровідних технологій, що швидко розвивається, є ZigBee, яка спочатку розроблялася як низькошвидкісна лінія зв'язку для об'єднання в мережу різних датчиків [1, 2]. Стосовно застосування ZigBee це можуть бути датчики охоронної і пожежної сигналізації, датчики збору телеметричної інформації, датчики медичних служб та інші.

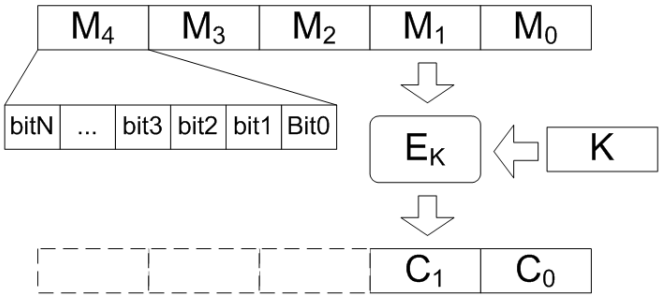

Рис. 1. Робота блочного шифрування

Принцип захисту інформації в телекомунікаційних системах.
Блочне шифрування для систем телекомунікацій

Блочне шифрування є різновидом симетричного шифрування, яке виконує дію над групами біт заданої довжини – блоками, що в загальному мають довжину блока в межах 64–256 біт.

Якщо відкритий текст має меншу довжину, ніж довжина заданого блока, то перед шифруванням цей блок доповнюється незначущою інформацією. Інакше кажучи, блочне шифрування являє собою підстановку на основі блочного алфавіту, який може бути моно- або ж поліалфавітним. Найбільшого застосування блочне симетричне шифрування набуло в передачі інформації мережт., в тому числі для систем пакетного обміну. Роботу блочного шифрування можна представити наступним чином [2, 3] (рис. 1).

Блочний шифр здатний зашифрувати одним ключем одне чи навіть декілька повідомлень, які в загальній сумі їх довжин будуть більшими, аніж довжина ключа шифрування. Однак постає питання про надійність такого шифрування. Ще одним з мінусів блочного шифрування є мала швидкість шифрування, в порівнянні з потоковим шифруванням.

Інтернет речей (IoT) існує менше 15 років, але розвивається з величезною швидкістю. Одночасно з нововведеннями виникають серйозні проблеми, що пов'язані з інформаційною безпекою. Алгоритми, на яких заснована сучасна криптографія, занадто складні та використовують багато ресурсів, а їх використання є складним завданням для малопотужних процесів Інтернету речей [2, 5, 6, 11].

Так, компанія Hewlett Packard провела дослідження, та в 2015 році виявила, що 70 % пристроїв IoT мають вразливості у безпеці своїх паролів, існують проблеми з шифруванням даних і з дозволом доступу і багато що інше. Так, IOT дав поштовх для розвитку нової гілки криптографії – Low Weight Cryptography, LWC. На сьогодні є близько 50 низькоресурсних алгоритмів, але в більшості вони абсолютно непридатні для використання. Розробники часто не мають змоги обрати алгоритм і чи зможе він працювати на певному пристрої. На рис. 2 показано деякі алгоритми та їх застосування.

SPONGENT	176-bit, на основі AES	алгоритми хешування	**Симетричні шифри**
PHOTON	64-bit, на основі AES		
LESAMNTA - LW	128-bit, на основі AES		
PRESENT	80, 128-bit	алгоритми блокового шифрування	
SPECK	64-256-bit		
CLEFIA	128-bit		
TRIVIUM	64-128-bit	алгоритм потокового шифрування	
Еліптичні криві	128-bit	алгоритм потокового шифрування	**Асиметричний шифр**

Рис. 2. Light Weight Cryptography (LWC) алгоритми (частина) [5–9]

У асиметричній криптографії використовуються наступні операції алгебри і алгоритми [10]:

- алгоритми перевірки числа на простоту;
- операції алгебри над великими числами: модульне множення, модульне піднесення до степеня, знаходження залишку від ділення, обчислення зворотного числа по модулю.

Найбільш трудомісткими операціями в асиметричній криптографії є операції модульного множення великих чисел і модульного піднесення до степеня великих чисел.

Розробка програмного забезпечення криптографічних операцій

Слід зазначити, що модульне піднесення до степеня складається з ітераційної послідовності

модульних множень. Якщо врахувати, що піднесення до степеня в криптографічних алгоритмах застосовується як безпосередньо, так і побічно (перевірка на простоту, обчислення зворотного числа по модулю та ін.), то стає очевидним, що головним завданням, яке необхідно реалізувати в арифметичному співпроцесорі, являється апаратне модульне множення великих чисел [9]:

$$M = a\,b\ (mod\ \ n),$$

де a, b – множники;
M – добуток;
n – модуль (просте число);
$a, b, M < n$; усі величини – великі позитивні цілі числа.

Визначення апаратних витрат на обрахунок операції

Виконаємо визначення обчислювальної складності. Фізичний час для обрахунку алфавітів для методу еліптичних кривих визначається як (рис. 3):

$$T_A(n) = \frac{n \cdot \tau_b + n^2 \cdot \tau_k}{f_{CPU}}, \text{ секунд,}$$

де n – розмірність очікуваного алфавіту;
τ_b – операційна складність обрахунку одної букви алфавіту;
τ_k – кількість операцій для виконання математичної операцій пошуку;
f_{CPU} – швидкодія процесора.

Передача інформації в системах IEEE 802.15.4 ZigBee відбувається із використанням блоків даних. Розмір корисної частини залежить від довжини службових полів. Версія стандарту 802.15.4b передбачає передачу інформаційного блоку довжиною 63 байти. Пізніша версія стандарту 802.15.4b дозволяє збільшити корисне навантаження фрейма, коли використовуються короткі адреси (16 біт замість 64). В цьому випадку об'єм даних дорівнюватиме 114 байтам. В такому випадку зростає обсяг корисних даних.

```
int pwr_mod(int a, int b, int m)
{
  int r=1;
  a%=m;
   while (b)
    {
      if (b&1)
        r=(r*a)%m;
      a=(a*a)%m;
      b>>=1;
    }
  return r;
}
```

Рис. 3. Реалізація алгоритму підведення в ступінь a^b

Для оцінювання потенційної працездатності для алгоритмів обробки даних за алгоритмом шифрування, потрібно визначити дії, що використовують найбільше процесорного часу. До таких дій відносяться операції підведення в ступінь. Для підведення в ступінь використовуються алгоритми, мета яких виконати дії над короткими числами, розрядність яких співрозмірна з розрядністю процесорної системи. Як можна побачити з рис. 3, підведення в ступінь складається з ряду дій – зсуву (множення) та ділення. Оцінка витрат часу наведена в табл. 1

Таблиця 1

Операційні витрати на обрахунок одної математичної операції

Тип процесора	τ_b, такти	τ_k, такти	f_{CPU}, МГц
	Ключ 32 біта		
8-біт, AVR RISC	24	140	24
32-біт, ARM RISC	7	42	200
	Ключ 64 біта		
8-біт, AVR RISC	180	2270	24
32-біт, ARM RISC	33	320	200

Результат моделювання часу для опрацювання розміру блока від 32 до 256 байт показано на рис. 3 та рис. 4., а детальні відомості – в таблиці 2.

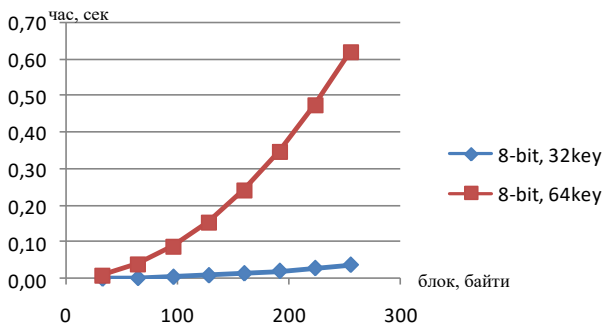

Рис. 4. Обчислення алфавітів для 8-бітного контролера

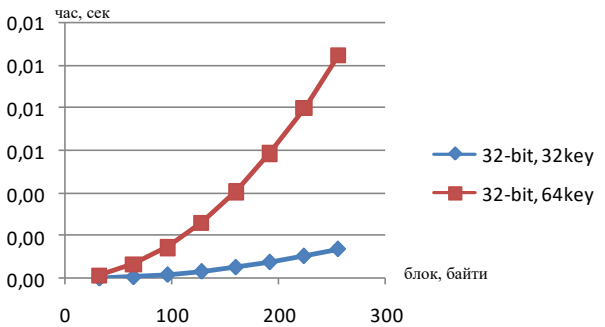

Рис. 5. Обчислення алфавітів для 32-бітного контролера

Як видно з рис. 3 та рис. 4 для 8-бітного контролера, час обрахунку одного значення алфавіту для 8-бітних контролерів є суттєвим значенням. В загальному видно, що застосування 32-бітної архітектури дозволяє отримати більш ніж 70 разову перевагу.

Таким чином, з таблиці 2 видно, що тільки для 8-бітного контролера з ключем в 64 біти не досягається можливість обробки блоків за 1 секунду.

Таблиця 2

Шифрування та підготовка даних для передачі згідно стандартів 802.15.4b та 802.15.4a

Розмір блоку шифрування, біти	802.15.4b		802.15.4a	
	Час підготовки блоку, 63 байт	Кількість блоків за 1 с	Час підготовки блоку, 127 байт	Кількість блоків за 1 с
8-бітний контролер, ключ 32 біти				
32	0,020808	48,05	0,041947	23,83
64	0,083012	12,04	0,167342	5,97
96	0,186611	5,35	0,376184	2,65
128	0,331605	3,01	0,668474	1,49
160	0,517994	1,93	1,044211	0,95
192	0,745779	1,34	1,503396	0,66
224	1,014959	0,98	2,046028	0,48
256	1,325533	0,75	2,672107	0,37
8-бітний контролер, ключ 64 біти				
32	0,336428	2,97	0,678197	1,47
64	1,34405	0,74	2,709435	0,36
96	3,022866	0,33	6,093714	0,16
128	5,372875	0,186	10,83103	0,092
160	8,394078	0,119	16,9214	0,059
192	12,08647	0,082	24,3648	0,041
224	16,45006	0,060	33,16124	0,030
256	21,48485	0,046	43,31073	0,023
32-бітний контролер, ключ 32 біти				
32	0,000749	1335,12	0,00151	662,30
64	0,002988	334,64	0,006024	166,00
96	0,006718	148,86	0,013542	73,84
128	0,011937	83,77	0,024064	41,55
160	0,018647	53,62	0,03759	26,60
192	0,026847	37,24	0,054121	18,47
224	0,036538	27,36	0,073655	13,57
256	0,047718	20,95	0,096194	10,39
32-бітний контролер, ключ 64 біти				
32	0,005695	175,58	0,011481	87,09
64	0,022745	43,96	0,045851	21,80
96	0,051148	19,55	0,103109	9,69
128	0,090906	11,00	0,183255	5,45
160	0,142018	7,04	0,28629	3,49
192	0,204484	4,89	0,412213	2,42
224	0,278304	3,59	0,561025	1,78
256	0,363478	2,75	0,732725	1,36

Враховуючи, що модулі ZigBee часто працюють в режимі інтервальної передачі даних, як-то передача в інтервалі хвилин або навіть годин, то в такому випадку встановлено, що 8-бітний контролер забезпечить роботу з шифрування та передачі її на основний модуль.

Висновки

Як показано в таблиці, блочний алгоритм підготовлює блоки даних, кратні 2^n байт. Але в стандартах 802.15.4a та 802.15.4b розмір блоку є 127 та 63 байти, а тому при передачі даних буде необхідно робити вирівнювання до більшого розміру. Так, при передачі 64 блоку даних буде використано один блок 127 байт – а, отже "доданих" байтів буде 63, майже 50%. Для блоку в 96 байт – зайвих потрібно 31 байт (25% від загального обсягу пакету). Тому обсяг корисних даних для стандарту ZigBee буде значно меншим ніж обсяг переданих даних. Якщо за стандартом 802.15.4 швидкість передачі даних складає до 250 кБіт/с, то реальна швидкість буде зменшена через наявність "доданих" байтів, що не несуть корисного навантаження.

Література

1. Стрельницкий А.А. Волновые каналы архитектурных сооружений. Усовершенствованная модель и новый эксперимент / А.А. Стрельницкий, А.Е. Стрельницкий, А.И. Цопа, В.М. Шокало // Радиотехника. Всеукраинский межведомственный научно-технический сборник. – 2007. – Выпуск 151. – С. 158–163.

2. Хоффман Л.Дж. Современные методы защиты информации / Л.Дж. Хоффман. – М. : Советское радио, 1980. – 164 с.

3. Панасенко С.П. Защита информации в компьютерных сетях / С. П. Панасенко // Мир ПК. – 2002 – № 2.

4. Хоффман Л.Дж. Современные методы защиты информации / Л. Д. Хоффман. – М. : Советское радио, 1980. – С. 87. – 164 с.

5. Ростовцев А.Г. Алгебраические основы криптографии / А. Г. Ростовцев. – СПб.: Мир и семья, Интерлайн, 2000. – С. 112-220.

6. Шнейер Б. Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке С / Б. Шнейер – М.: Триумф, 2002. – С. 12. – 216 с.

7. Чмора А. Л. Современная прикладная криптография. – М.: "Гелиос АРВ", 2001. – 256 с.

8. Соловьев Ю.П. Эллиптические кривые и современные алгоритмы теории чисел / Ю. П. Соловьев, В. А. Садовничий, Е.Т. Шавгулидзе. – М. : Ижевск, 2003. – С. 60–92.

9. Lopez J. Fast multiplication on elliptic curves over GF(2n) without precomputation / J. Lopez, R. Dahab. – Lecture Notes in Computer Science. – 2000. – № 1965. – P. 317–327.

10. Василенко О.Н. Теоретико-числовые алгоритмы в криптографии / Василенко О.Н. – М. : МЦНМО, 2003. – 325 с.

11. Горященко К.Л. Впровадження технологій PLC / К.Л. Горященко, О.П. Войтюк, С.Л. Кушнірук, О.В. Шевчук // Вісник Хмельницького національного університету. Технічні науки. – 2017. – № 2. – С. 250–253.

References

1. Strelnickij A.A. Volnovye kanaly arhitekturnyh sooruzhenij. Usovershenstvovannaya model i novyj eksperiment / A.A. Strelnickij, A.E. Strelnickij, A.I. Copa, V.M. Shokalo // Radiotehnika. Vseukrainskij mezhvedomstvennyj nauchno-tehnicheskij sbornik. – 2007. – Vypusk 151. – S. 158–163.

2. Hoffman L.Dzh. Sovremennye metody zashity informacii / L.Dzh. Hoffman. – M. : Sovetskoe radio, 1980. – 164 s.

3. Panasenko S.P. Zashita informacii v kompyuternyh setyah / S. P. Panasenko // Mir PK. – 2002 – № 2.

4. Hoffman L.Dzh. Sovremennye metody zashity informacii / L. D. Hoffman. – M. : Sovetskoe radio, 1980. – C. 87. – 164 s.

5. Rostovcev A.G. Algebraicheskie osnovy kriptografii / A. G. Rostovcev. – SPb.: Mir i semya, Interlajn, 2000. – S. 112-220.

6. Shnejer B. Prikladnaya kriptografiya. Protokoly, algoritmy, ishodnye teksty na yazyke C / B. Shnejer – M.: Triumf, 2002. – S. 12. – 216 s.

7. Chmora A. L. Sovremennaya prikladnaya kriptografiya. – M.: "Gelios ARV", 2001. – 256 s.

8. Solovev Yu.P. Ellipticheskie krivye i sovremennye algoritmy teorii chisel / Yu. P. Solovev, V. A. Sadovnichij, E.T. Shavgulidze. – M. : Izhevsk, 2003. – C. 60–92.

9. Lopez J. Fast multiplication on elliptic curves over GF(2n) without precomputation / J. Lopez, R. Dahab. – Lecture Notes in Computer Science. – 2000. – № 1965. – P. 317–327.

10. Vasilenko O.N. Teoretiko-chislovye algoritmy v kriptografii / Vasilenko O.N. – M. : MCNMO, 2003. – 325 s.

11. Horiashchenko K.L. Vprovadzhennia tekhnolohii PLC / K.L. Horiashchenko, O.P. Voitiuk, S.L. Kushniruk, O.V. Shevchuk // Herald of Khmelnytskyi National University. – 2017. – № 2. – S. 250–253.

Надійшла / Paper received : 21.10.2020 Надрукована/Printed :27.11.2020