**УДК 519.688**

**К.т.н., доцент Тесленко О.К., студент Делавар М.Т.**

**Національний технічний університет України**
**«Київський політехнічний інститут»**

# ОПТИМІЗАЦІЯ ПРОГРАМНИХ РЕАЛІЗАЦІЇ АЛГОРИТМІВ КРИПТОГРАФІЧНИХ ПЕРЕТВОРЕНЬ

## Abstract

***Alexander C. Teslenko, assoc. prof., PhD; Tariq Dellawar, student***
***Program realization optimization of cryptographic transformation algorithms***

*This paper concerns the task of program realization optimization of cryptographic transformation algorithms. The regimes of application of symmetric-key encryption algorithms are discussed and performances of each of them are evaluated. Most attention in this paper is paid to performance issues. The methods to increase performance of algorithms on multi-core processors are studied and the results are analyzed. The ways for further researches are proposed.*

## Вступ

У сучасному суспільстві зростають вимоги до продуктивності обчислювальних систем, змінюються і вимоги до все більш складних додатків. Одним з можливих рішень є розробка і застосування систем з масовим паралелізмом - кластерів, Grid-систем, багатопроцесорних комплексів, систем на багатоядерних процесорах [1,2].

З появою складних автоматизованих систем управління, пов'язаних з автоматизованим введенням, зберіганням, обробкою і виведенням інформації, широке використання обчислювальних мереж приводить до того, що з'являються великі можливості для несанкціонованого доступу до інформації. В вирішенні проблеми захисту інформації криптографічні перетворення набувають все більшого значення. Існує цілий ряд національних та міжнародних стандартів криптографічних перетворень, серед яких до найбільш перспективних відносять стандарт блочного симетричного шифрування AES (Advanced Encryption Standard) [3] Враховуючи постійно зростаючий об’єм даних для шифрування, велике практичне значення мають дослідження по підвищенню швидкості криптографічних засобів, зокрема шляхом використання систем з масовим паралелізмом.

## Постановка задачі

Провести аналіз методів збільшення продуктивності роботи алгоритму блочного симетричного шифрування AES. Запропонувати та дослідити методи підвищення продуктивності програмних реалізацій алгоритму AES для обчислювальних систем на базі багатоядерних процесорів.

## Режими роботи блочного симетрично шифрування

Алгоритм блочного симетричного шифрування AES може застосовуватися в одному з наступних п'яти режимів: ECB, CBC, CFB, OFB, CTR.

1. Режим ECB (Electronic Codebook) передбачає, що кожен блок відкритого тексту замінюється на блок зашифрованого тексту, причому однакові блоки відкритого тексту дають однакові блоки зашифрованого тексту.
2. Режим CBC (Cipher Block Chaining) передбачає складання блоку відкритого тексту з попереднім блоком зашифрованого тексту за допомогою побітової операції XOR.
3. Режим CFB (Cipher Feedback) кожен одержаний блок зашифрованого тексту надходить на вхід при шифруванні наступного блоку.
4. Режим OFB (Output Feedback) передбачає наявність так званого ініціюючого блоку, який надходить на вхід в процедуру шифрування, а отриманий зашифрований блок знову проходить процедуру шифрування, і т. д. Отримані в ході цього блоки складаються з блоками відкритого тексту за допомогою побітової операції XOR .
5. Режим CTR (Counter) передбачає наявність лічильника. Лічильник - це 128-бітове число, яке перед шифруванням ініціалізується випадковим значенням і для кожного наступного блоку відкритого тексту приймає нове значення, відмінне від всіх попередніх. Значення лічильника шифруються і складаються з блоками відкритого тексту за допомогою побітової операції XOR. Початкове значення лічильника передається разом з зашифрованими даними і використовується при розшифруванні.

Для вибору режиму застосування алгоритму AES, найбільш прийнятного для паралельної реалізації, використовувався наступний набір критеріїв.

*Можливість розпаралелювання процедури шифрування (розшифрування)*. Цей критерій означає можливість шифрування (розшифрування) двох і більше блоків одночасно.

*Простота реалізації*, що означає, що при шифруванні і розшифруванні використовується тільки пряме криптографічне перетворення, а обернене не використовується.

*Можливість попередніх обчислень*. Цей критерій означає, що можна почати обчислення до надходження відкритого тексту.

У Табл. 1 наведено оцінки режимів з даного набору критеріїв. У цій таблиці «+» означає, що режим відповідає критерію, «-» - ні.

Таблиця 1

Оцінка режимів застосування алгоритму симетрічного шифрування

| Критерій | ECB | CBC | CFB | OFB | CTR |
|---|---|---|---|---|---|
| Можливість розпаралелювання процедури шифрування | + | - | - | - | + |
| Можливість розпаралелювання процедури розшифрування | + | + | - | - | + |
| Простота реалізації | - | - | + | + | + |
| Можливість попередніх обчислень | - | - | - | + | + |

Таким чином, відповідно до проведеного аналізу режим CTR є найбільш прийнятним для паралельних обчислень.

**Реалізація паралельної версії алгоритму шифрування**

Паралельна версія алгоритму складається з головного потоку, який називається "Main" і другорядного "Secondary" де виконуються функції, які можуть бути виконані паралельно.

Основні функції головного потоку "Main":

- читання і запис блоків;
- формування і відправка завдань;
- управління потоками "Secondary".

Основні функції потоку "Secondary":

Потік "Secondary" отримує блоки відкритого тексту від потоку "Main", шифрує їх і передає назад результат. Потік "Secondary" може знаходитися в одному з двох станів: «вільний» і «зайнятий». Стан «вільний» означає,

що потрібно відправити завдання для виконання на даному потоці, а «зайнятий» — означає, що потік зайнятий виконанням отриманого завдання.

Для оцінки ефективності запропонованого методу було поставлено завдання, з'ясувати яке прискорення криптографічного перетворення дає запропонований метод, а також визначити розширюваність алгоритму.

*Прискоренням* називається величина

$$S_p(n) = \frac{T_1(n)}{T_p(n)}$$

де n — розмір завдання, $T_1$ — час рішення на одному потоці, $T_p$ — час рішення на *p* потоках [2].

*Розширюваність* – це залежність часу виконання алгоритму від розміру задачі. Реалізація алгоритму вважається розширюваною якщо час виконання лінійно залежить від розміру задачі.

**Результати експериментів**

Проведені експерименти показали наступне. При шифруванні файлів розміром до 1Гбайт на двох ядерному процесорі час шифрування зменшився на 45% в порівнянні з виконанням того ж завдання послідовно.

Після проведення шифрування файлів різних розмірів було встановлено лінійну залежність часу шифрування від розміру файлу. Таким чином запропонований метод має властивості розширюваності.

**Висновки**

Запропонований метод дозволяє суттєво підвищити швидкість криптографічних перетворень по алгоритму AES для файлів різного розміру.

**Література**


1. *Dongarra J.* An Overview of High Performance Computing and Challenges for the Future // High Performance Computing for Computational Science - VECPAR 2008. _ Springer Berlin, 2008.
2. *Гергель В.П.* Теория и практика параллельных вычислений. ИНТУИТ, Бином. Лаборатория знаний, 2007. 424 с.
3. Specification for the Advanced Encryption Standard (AES). — FederalInformation Processing Standards Publication 197. — 2001.