УДК 004.56.5(043.2)

Глинська М.Л.[1], Рибалка І.М.[2]

[1]Галицький коледж ім.. В.Чорновола

[2]Західноукраїнський національний університет

ДОСЛІДЖЕННЯ АЛГОРИТМУ ШИФРУВАННЯ RC6

Вступ. RC6 - симетричний блоковий криптографічний алгоритм, похідний від алгоритму RC5. Був створений Роном Ривестом, Меттом Робшау і Реєм Сіднеєм для задоволення вимог конкурсу Advanced Encryption Standard (AES). Алгоритм був одним з п'яти фіналістів конкурсу, був також представлений NESSIE і CRYPTREC. Алгоритм є власністю (пропрієтарним), і запатентований RSA Security.

Варіант шифру RC6, заявлений на конкурс AES, підтримує блоки довжиною 128 біт і ключі довжиною 128, 192 і 256 біт, але сам алгоритм, як і RC5, може бути налаштований для підтримки більш широкого діапазону довжин як блоків, так і ключів (від 0 до 2040 біт). RC6 дуже схожий на RC5 за своєю структурою і також досить простий в реалізації.

Є фіналістом AES, однак одна з примітивних операцій - операція множення, повільно виконувана на деякому обладнанні і ускладнює реалізацію шифру на ряді апаратних платформ, що виявилося проблемою, на системах з архітектурою Intel IA-64 також реалізована досить погано. В даному випадку алгоритм втрачає одну зі своїх ключових переваг - високу швидкість виконання, що стало причиною для критики і однією з перепон для обрання в якості нового стандарту. Однак, на системах з процесором починаючи з Pentium II, Pentium Pro, Pentium III, PowerPC і ARM алгоритм RC6 випереджає переможця - Rijndael.

Алгоритм RC6 є продовженням криптоалгоритму RC5, розробленого Рональдом Ривестом. RC5 був незначно змінений для того, щоб відповідати вимогам AES по довжині ключа і розміром блоку. При цьому алгоритм став ще швидше, а його ядро, успадковане від RC5, має солідний запас досліджень.

Мета: Дослідження систем шифрування RC6 з метою можливостей його реалізації та вдосконалення.

1. Мережа Фейстеля алгоритму RC6

Алгоритм є мережею Фейштеля з 4 гілками змішаного типу: в ньому два парних блоки використовуються для одночасної зміни вмісту двох непарних блоків. Потім проводиться звичайне для мережі Фейштеля зрушення на одне машинне слово, що змінює парні і непарні блоки місцями.

Сам алгоритм роботи мережі Фейштеля досить простий і зображений на рисунку 1. Розробники алгоритму рекомендують при шифруванні використовувати 20 раундів мережі, хоча в принципі їх кількість не регламентується. При 20 повторах операції шифрування алгоритм має найвищу швидкість серед 5 фіналістів AES.

Так само, як і RC5, RC6 - повністю параметризований алгоритм шифрування. Для специфікації алгоритму з конкретними параметрами, прийнято позначення RC6-w / r / b , де:

- w - довжина машинного слова в бітах.
- r - число раундів.
- b - довжина ключа в байтах. Можливі значення 0..255 байт.

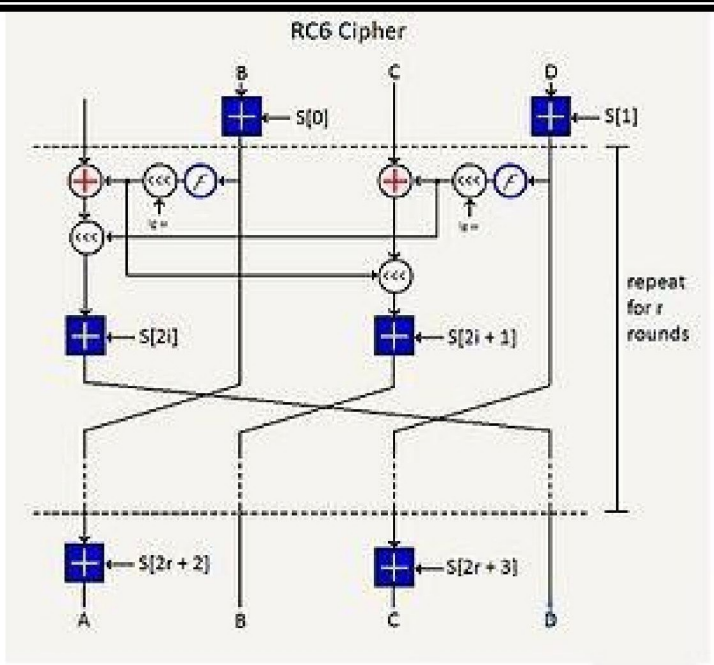

Рисунок 1 - Мережа Фейштеля

Перетворення T (x) виконується так: T (X) = (X * (X + 1)) mod 2N. Воно використовується в якості нелінійного перетворення з хорошими показниками перемішування бітового значення вхідної величини.

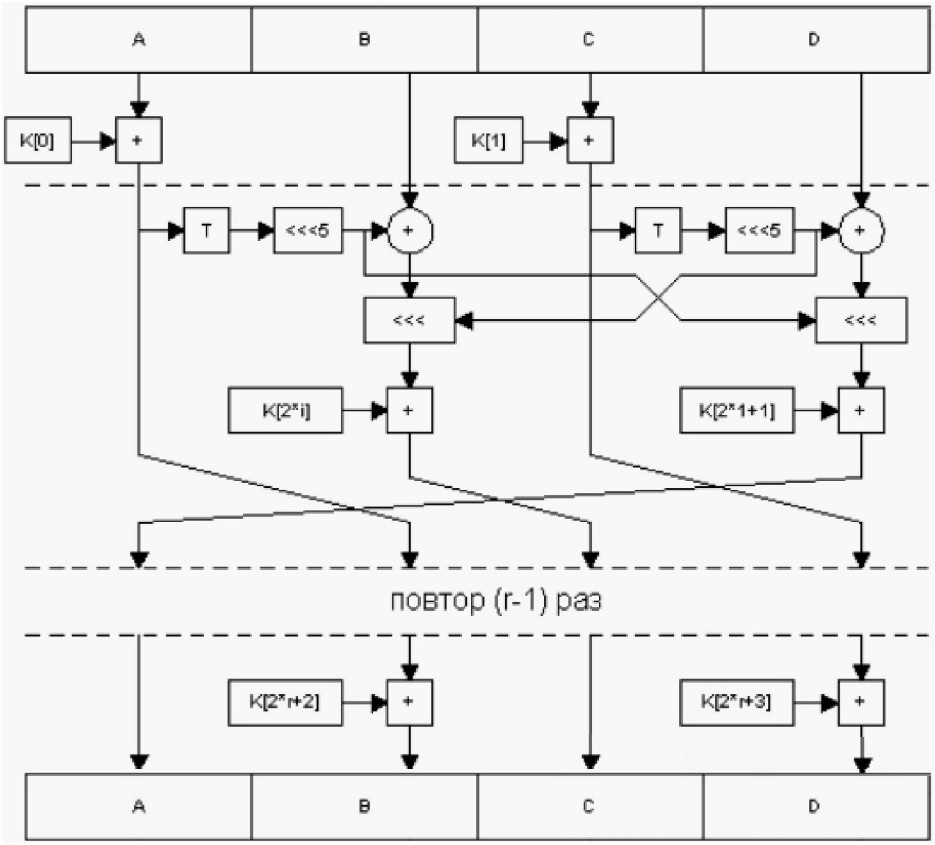

Рисунок 2 - Узагальнена схема алгоритму шифрування методом RC6

Для того щоб відповідати вимогам AES, блоковий шифр повинен звертатися з 128-бітовими блоками. Так як RC5 - виключно швидкий блоковий шифр, його розширення, щоб працювати з 128-бітовими блоками призвело б до використання двох 64-бітових робочих регістрів.

2. Безпека алгоритму RC6

Варіант алгоритму RC6, який був заявлений на AES, як уже було сказано, підтримує блоки довжиною 128 біт і ключі довжиною 128, 192 і 256 біт, а також містить 20 раундів. Тобто RC6-128 / 20 / *b*, де *b* = 128,192 або 256 біт. Відносно такого алгоритму ніяких атак не було виявлено. Були виявлені атаки тільки проти спрощених версій алгоритму, тобто алгоритму зі зменшеною кількістю раундів.

Вважається, що найкращий варіант нападу на RC6, доступний для криптоаналітика, є повним перебором *b*-байтового ключа шифрування (або розширений ключовий масив *S* [0, ..., 43], коли наданий користувачем ключ шифрування особливо довгий). Дон Копперсміт зауважив, що за рахунок використання великої кількості пам'яті і попереднього обчислення можна організувати атаку «meet-in-the-middle», щоб відновити розширений ключовою масив *S* [0, ..., 43]. Більш просунуті атаки, такі як диференційний і лінійний криптоаналіз, здійснювані на версіях шифру з невеликою кількістю раундів, складно здійснювати для нападу на повний шифр rc6 з 20 раундами. Складність полягає в тому, що важко знайти хороші повторювані особливості або лінійні наближення, з якими могла б бути здійснена атака.

Проблемою є встановлення відповідних цілей для безпеки проти цих більш просунутих атак. Щоб досягти успіху, ці атаки типово вимагають великої кількості даних, і отримання блоків відомих або обраних пар зашифрованого \ відкритого тексту - завдання відмінне від спроби повернути один ключ з можливих. Варто зауважити, що з шифром, що працюють із швидкістю один терабіт в секунду (тобто, шифруючи дані зі швидкістю 10^{12} біт / сек), час, необхідний для 50 комп'ютерів, що працюють паралельно, щоб зашифрувати блоків даних, становить понад рік; розшифрувати 2^{128} блоків даних - більше ніж: 98 000 років; і зашифрувати 2^{128} блоків даних складає більше ніж 10^{19} років.

Дослідження RC5 не проявили слабкостей в установці ключа. Це призвело до використання того ж процесу установки ключа і в RC6. Процес перетворення ключа, наданого користувачем, до таблиці ключів, здається, добре змодельований псевдовипадковим процесом. Таким чином, в той час як немає доказу, що ніякі два ключа не призводять до однієї і тієї ж таблиці ключів, це, здається, дуже малоймовірно. Це можна оцінити як вірогідність того, що існують два 256-бітових ключа, що призводять до однієї і тієї ж таблиці 44, 32-розрядних ключів, тобто приблизно 10^{-270} .

Ми можемо підсумувати наші висновки по безпеці RC6 наступним чином:

- Найкращою атакою на RC6 є повний перебір для забезпеченого користувачем ключа шифрування.

- Вимоги до даних, щоб організувати більш складні атаки на RC6, такі як диференційний і лінійний криптоаналіз, перевищують доступні дані.

Важливим критерієм резерву безпеки є максимальне число раундів, при якому можлива атака. Це можливо для 12-, 14- і 15- раундових варіантів RC6.

3. Оцінка апаратних засобів для реалізації алгоритму RC6

Для більшості додатків впровадження RC6 в програмне забезпечення - ймовірно, кращий вибір. Примітивні операції RC6 (додавання, віднімання, множення, що виключає операцію або та зсув) дуже добре підтримуються сучасними мікропроцесорами і тому при розробці цих процесорів вигідно це враховувати.

Однак, в деяких випадках необхідно мати RC6 у вигляді вбудованої схеми. Тоді можна було б досягти максимальної швидкості або об'єднати інші функції навколо RC6. Оскільки RC6 використовує примітивні операції, описані вище, то можна використовувати переваги існуючої перевірки при розробці схемних модулів для реалізації цих примітивних операцій.

Наприклад, якщо реалізувати RC6, використовуючи технології, засновані на матрицях логічних елементів, то це не принесе бажаних переваг через велику кількість додаткового коду, якимй потрібно буде написати для розробки схеми множення. Реалізація на базі такої технології значно поступається реалізації на базі процесора. Але це не типова ситуація і можна легко зпроектувати схему множення, яка буде використовуватися в якості підмодуля.

З 20 раундами на блок час шифрування приблизно дорівнює 100 наносекунд для кожного блоку, забезпечуючи передбачувану швидкість передачі даних приблизно 1.3 Гбіт / сек.

Як випливає з опису алгоритму, RC6 - дуже компактний. Дійсно, реалізація алгоритму RC6 на Асемблері для мікропроцесора Intel Pentium Pro може бути здійснена в меншій реалізації ніж 256 байтах коду для кожної з задач:

1. установки ключа,
2. блоку шифрування,
3. блоку дешифрування.

На відміну від багатьох інших алгоритмів шифрування RC6 не використовує довідкові таблиці під час шифрування. Це означає, код RC6 і дані можуть міститися в сучасній кеш пам'яті і тим самим економити місце в пам'яті.

З огляду на, що RC6 повністю параметризується, і що він може бути ефективно і компактно здійснений, шифр здається особливо універсальним.

Висновок. RC6 надає користувачеві велику гнучкість щодо розміру ключа шифрування, числа раундів і розміру слова основного обчислювального модуля.

У той час як RC6, представлений для розгляду на AES, базується на використання 32-розрядних слів (розмір блоку 128 біт), майбутня потреба ринку потребує розширення RC6 для інших розмірів блоку. На передньому плані представляють розміри блоку в 256 біт, які використовували б розмір слова 64 біт і продуктивність, пропоновану наступним поколінням системної архітектури. Також відзначимо, що структура RC6 дозволяє експлуатувати певний ступінь паралелізму в підпрограмах розшифрування і шифрування. Наприклад, обчислення t і u в кожному раунді може бути обчислено паралельно, як і оновлення A і C. Оскільки процесори розвиваються в напрямку збільшення кількості внутрішнього паралелізму (наприклад, з переміщенням до суперскалярної архітектурі), реалізації RC6 повинні продемонструвати велику продуктивність.

Перелік використаних джерел.

1. Ryabko B.Ya., Monarev V.A. Using information theory approach to randomness testing//Journal of Statistical Planning and Inference, 2005. Vol. 133. № 1. PP. 95-110.

2. Knudsen L., Meier W. Correlations in RC6 with a reduced number of rounds//FSE 2000. LNCS 1978(2000). Springer-Verlag. P. 94-108.

3. Biham E., Dunkelman O., Keller N. A New Attack on 6-Round IDEA//Lecture Notes in Computer Science. Berlin, Heidelberg: Springer-Verlag, 2007. V. 4593. P. 211-224.