computing@tanet.edu.te.ua
www.tanet.edu.te.ua/computing

ISSN 1727-6209
International Scientific Journal of Computing

БАЗОВІ СТРУКТУРИ ОПЕРАЦІЙНИХ ПРИСТРОЇВ ХЕШУВАННЯ ДЛЯ ПРОЦЕСОРІВ ПІДТРИМКИ ПРОТОКОЛУ IPSEC

Коркішко Л. М. [1)], Коркішко Т. А. [2)], Шевчук Р. П. [3)]

[1)] асистент кафедри безпеки інформаційних технологій, lk@tanet.edu.te.ua
[2)] к.т.н., старший викладач кафедри комп'ютерних наук, tko@tanet.edu.te.ua
[3)] студент, rsh@tanet.edu.te.ua
Інститут комп'ютерних інформаційних технологій,
Тернопільська академія народного господарства,
вул. Львівська, 11, м. Тернопіль, 46004

Анотація: *У даній роботі розглянуто базові структури операційних пристроїв хешування для процесорів підтримки протоколу IPSec. Розглянуто особливості алгоритмів хешування MD5 і SHA-1. Виділено базові операції алгоритмів та на їх основі розглянуто структури операційних пристроїв хешування. Виділено ряд граф-алгоритмічних операційних пристроїв. Отримано аналітичні вирази, які описують часові характеристики цих пристроїв. Використовуючи результати синтезу базової операції алгоритму SHA-1 на програмовану логічну інтегральну схему, отримано графіки залежностей часових параметрів операційних пристроїв та виділено області їх доцільного використання.*

Ключові слова: – *Протокол IPSec, спеціалізовані процесори, потоковий граф, хешування, алгоритми MD5, SHA-1.*

1. ВСТУП

Протокол IPSec [1,2] використовується для забезпечення цілісності, автентичності та конфіденційності даних, що передають незахищеними комп'ютерними мережами. Головною перевагою IPSec, яка зумовила його широке використання, є можливість шифрування і/або автентифікування всієї інформації, яка передається на рівні інтернет-протоколу. Однак, невисока продуктивність роботи програмованих процесорів, ресурси яких використовуються для реалізації протоколу IPSec, та втрати, спричинені часом накопичення пакета для передавання даних, зменшують продуктивність віртуальних (приватних) мереж, побудованих на основі протоколу IPSec. Тому постає задача створення спеціалізованих процесорів хешування для підвищення продуктивності обробки даних.

Одним з можливих шляхів розв'язання цієї задачі є створення спеціалізованих процесорів IPSec для виконання базових криптографічних алгоритмів протоколу IPSec. В основу побудови цих процесорів закладено принцип апаратного відображення структури виконуваного алгоритму на операційні пристрої тракту обробки даних. Протокол IPSec передбачає обробку даних алгоритмами симетричного блокового шифрування і хешування. Структури операційних пристроїв для виконання алгоритмів симетричного блокового шифрування докладно досліджено у [3]. Тому дана робота присвячена аналізу шляхів апаратного відображення алгоритмів хешування та створенню базових структур операційних пристроїв хешування для спеціалізованих процесорів підтримки протоколу IPSec.

2. ПОСТАНОВКА ЗАДАЧІ

До переліку алгоритмів хешування, які використовуються у протоколі IPSec входять: MD5 [4], SHA-1 [5]. Ці алгоритми використовуються вузький набір елементарних операцій: зсув на задану кількість біт, додавання за модулем 2^{32}, логічні операції, заміна частин блоку згідно з таблицею чи наперед визначеною функцією. Алгоритми хешування структурно організовані як послідовність різного числа раундів (табл. 1).

Таблиця 1. Параметри алгоритмів хешування

Алгоритм	MD5	SHA-1
Кількість раундів	65	81
Розмір буферу, біт	512	
Розмір хеш-значення, біт	128	160

Виконання елементарних операцій перелічених алгоритмів хешування на універсальних процесорах приводить до значних часових затрат на виконання цих операцій, і, як наслідок, зниження продуктивності обробки даних. Це зумовлено невідповідністю систем команд процесора та режимів адресації використовуваним операціям [6]. Тому зусилля фірм-виробників сконцентровані на розробці спеціалізованих процесорів захисту інформації, орієнтованих на реалізацію криптографічних алгоритмів протоколу IPSec. Серед цих спеціалізованих процесорів можна виділити: VMS747 [7], SafeXcel-2141 [8], NSP2000 [9], Hipp 7814, Hipp 7854, Hipp 7955 [10], MPC185TS/D, MPC184TS/D [11]. Основною перевагою спеціалізованих процесорів є збільшення продуктивності обробки даних в порівнянні з програмованими процесорами. Разом з тим, побудова операційних пристроїв спеціалізованих процесорів зумовлює необхідність дослідження шляхів апаратної реалізації алгоритмів хешування та формального опису їх характеристик. Отримавши аналітичні вирази, що пов'язують часові характеристики операційних пристроїв з їх параметрами, та відповідні аналітичні вирази для операційних пристроїв симетричного блокового шифрування [3], можна сформулювати та розв'язати задачу оптимізації апаратних засобів реалізації криптографічних алгоритмів IPSec за критеріями продуктивності, апаратних затрат, складністю пристроїв керування тощо.

3. БАЗОВІ СТРУКТУРИ І ЧАСОВІ ХАРАКЕТРИСТИКИ ОПЕРАЦІЙНИХ ПРИСТРОЇВ ХЕШУВАННЯ

Алгоритми хешування MD5 і SHA-1 можна представити у вигляді двох пов'язаних процедур: обчислення розпису повідомлення у вхідному буфері та обчислення хеш-значення (рис. 1).

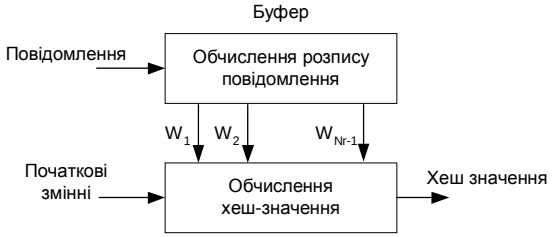

Рис. 1 – Структура алгоритмів хешування.

Для дослідження шляхів апаратної реалізації алгоритмів хешування представимо ці алгоритми у ярусно-паралельній формі. Це дасть змогу розкрити їх базові обчислювальні та структурні характеристики [12, 13]. На рис. 2 показано потоковий граф алгоритму хешування для обробки одного буферу даних алгоритму хешування. Він складається з Nr ярусів, з яких $Nr-1$ ярусів є структурно-подібними. Під ярусом потокового графу будемо розуміти такий набір функціональних операторів та операторів передачі даних, який забезпечує виконання одного раунду алгоритму. На кожен функціональний оператор $ФО_1, ФО_2,...,ФО_{Nr-1}$ подаються з буфера значення $W_1, W_2,...,W_{Nr-1}$, над якими виконуються базові операції алгоритму хешування. Останній функціональний оператор $ФО_{Nr}$ виконує додавання за модулем 2^{32} значення $ФО_{Nr-1}(W_{Nr-1})$ і змінної зчеплення CV_q.

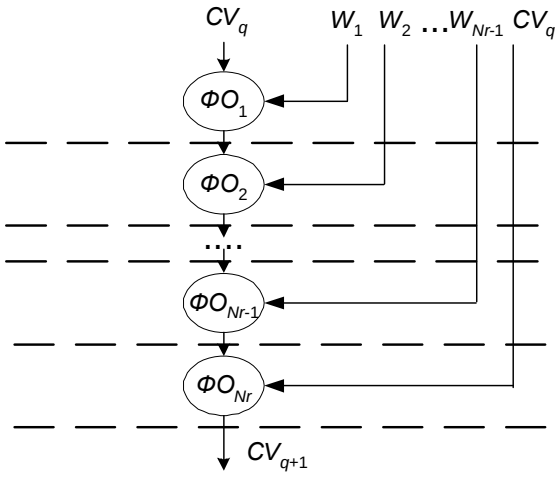

Рис. 2 – Потоковий граф алгоритмів хешування.

Якщо розглянутий вище потоковий граф апаратно відобразити на комбінаційній матриці, де операції $ФО_1, ФО_2,...,ФО_{Nr}$ виконуються на основі комбінаційних схем (КС), кожна з яких виконує операції одного ярусу, а після $ФО_{Nr}$ розміщено регістр даних то отримаємо граф-алгоритмічний операційний пристрій (ГАОП) [12]. Розглянемо часові характеристики ГАОП.

Час хешування даних з одного буферу алгоритму із використанням ГАОП буде рівний:

$$T_{ГАОП} = (Nr\text{-}1)*T_{КС1}+T_{КСNr}+T_{Рг}, \quad (1)$$

де $T_{КС1}$ – час обробки даних в комбінаційній схемі, що реалізує $ФО_1,...,ФО_{Nr-1}$; $T_{КСNr}$ – час обробки даних в комбінаційній схемі, що реалізує $ФО_{Nr}$; $T_{Рг}$ – час запису даних у вихідний регістр.

Оскільки процесори хешування орієнтують на обробку потоків даних, то доцільно розглянути принцип обробки даних, який передбачає суміщення в часі виконання функціональних операторів алгоритму над різними даними. Одним із можливих варіантів тут є конвеєризація ГАОП і створення конвеєрних ГАОП (КГАОП) [12], структура яких базується на апаратному відображені повного потокового графа алгоритму хешування з використанням всіх можливостей часового і просторового розпаралелення обробки. КГАОП складається з Nr послідовно з'єднаних комбінаційних схем, що реалізують відповідні раунди алгоритму хешування, розділених конвеєрними регістрами. Для алгоритму хешування MD5 будемо вважати, що Nr_{MD5}=65, причому 64 раунди є структурно-подібними, а останній раунд виконує операцію додавання за модулем 2^{32}. Аналогічно, для алгоритму хешування SHA-1, Nr_{SHA-1}=81. Послідовність комбінаційних схем розбита конвеєрними регістрами із врахуванням вимоги $(Nr-1) \bmod Npp = 0$, де Npp – кількість конвеєрних регістрів. При цьому, кількість комбінаційних схем між конвеєрними регістрами буде визначатися числом $Nksr = (Nr-1)/Npp$ (рис. 3).

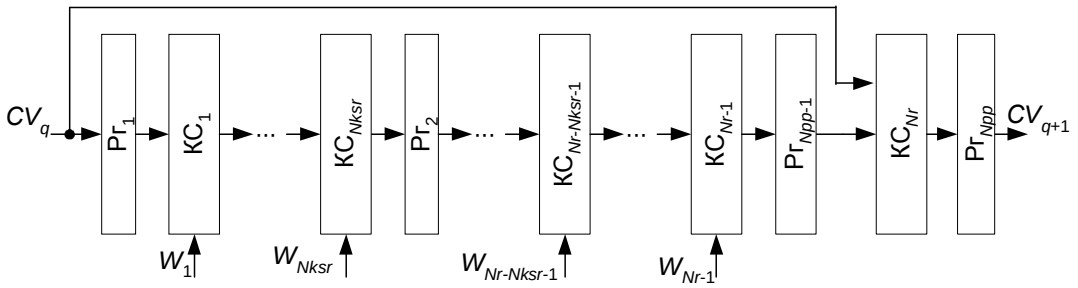

Рис. 3 – Структура КГАОП для виконання алгоритмів хешування.

Основні часові характеристики КГАОП хешування: такт роботи - $t_{КГАОП}$ та час обробки даних - $T_{КГАОП}$ знаходимо з виразів

$$t_{КГАОП} = Nksr*T_{КС1}+T_{Рг}, \quad (2)$$

$$T_{КГАОП} = Npp* t_{КГАОП} +T_{КСNr}+T_{Рг}. \quad (3)$$

Інший варіант побудови операційного пристрою хешування полягає у ітераційному виконанні обчислень згідно, тобто створення ітераційного ГАОП (ІГАОП) (рис. 4). Тут комбінаційні схеми реалізують проекцію потокового графу алгоритму хешування. Розгортання структури графу проводиться послідовно у часі. Характерною особливістю цієї структури є використання комутатора даних (Ком) для забезпечення ітераційної обробки даних, двох конвеєрних регістрів ($Рг_1$ і $Рг_2$), перший з яких призначений для ітераційного виконання Nr-1 раундів алгоритму хешування, а другий – для зберігання проміжного результату хешування.

Між комутатором і конвеєрним регістром $Рг_1$ розташовують таку кількість КС $Nksr$, щоб виконувалася рівність $(Nr\text{-}1) \bmod Nksr = 0$.

В ІГАОП обчислення проводиться шляхом багатократного проходження векторів даних $(W_1, W_2,...,W_{Nr-1})$ через операційний пристрій. Такт ІГАОП $t_{ІГАОП}$ та час обробки одного буферу даних $T_{ІГАОП}$ можна знайти з виразів:

$$t_{ІГАОП} = Nksr*T_{КС1}+T_k+T_{Рг}, \quad (5)$$

$$T_{ІГАОП} = (Nr\text{-}1)* t_{ІГАОП}/Nksr+T_{КСNr}+T_{Рг}. \quad (6)$$

Проміжне місце між ітераційними та конвеєрними ГАОП займає ітераційно-конвеєрний ГАОП (ІКГАОП). Так, як згідно з алгоритмом хешування блоки даних обробляються за фіксовану кількість структурно подібних раундів, то ітераційно-конвеєрний операційний пристрій повинен містити дві чи декілька комбінаційних схем, що реалізують проекцію функціональних операторів потокового графу (раундів), через які блоки даних проходять задану кількість разів та надходять на вихід (рис.

5).

Кількість конвеєрних регістрів для цієї структури операційного пристрою визначається числом *Npp*, яке визначається з нерівності $1 < Npp < (Nr-1)/Nksr$, де *Nksr* – кількість реалізованих комбінаційних схем між конвеєрними регістрами, $(Nr-1) \bmod Npp = 0$. При інших значеннях числа конвеєрних регістрів ускладнюється керування операційним пристроєм внаслідок необхідності організації асинхронної роботи складових операційного пристрою та пропуску чи обходу комбінаційних схем, які розташовані ближче до останнього конвеєрного регістра.

Дані у операційний пристрій подаються у вигляді матриці, де кількість рядків матриці визначається кількістю тактів для обробки одного буферу, а кількість стовпців – число реалізованих комбінаційних схем і конвеєрних регістрів.

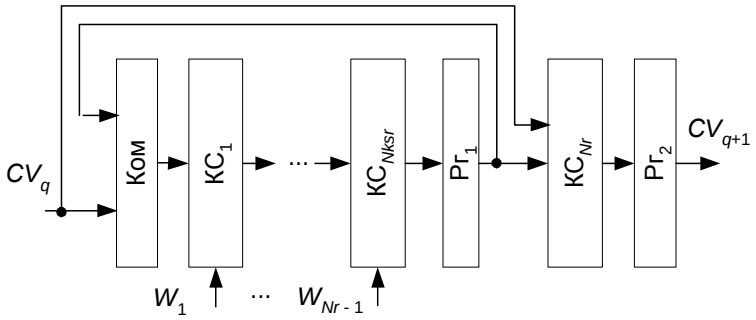

Рис. 4 – Структура ітераційного ГАОП для виконання алгоритмів хешування.

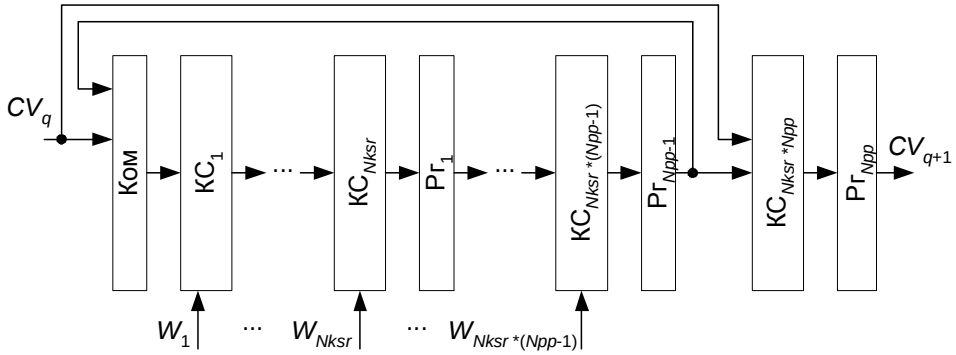

Рис. 5 – Структура ітераційно-конвеєрного ГАОП для виконання алгоритмів хешування.

Такт роботи та час обробки одного буферу даних для ІКГАОП складає відповідно:

$$t_{IKГAOП} = Nksr*T_{KC1}+T_k+T_{Pг}, \quad (6)$$

$$T_{IKГAOП} = (Nr\text{-}1)*t_{IГAOП}/Npp+T_{KCNr}+T_{Pг}. \quad (7)$$

4. ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ БАЗОВИХ СТРУКТУР ОПЕРАЦІЙНИХ ПРИСТРОЇВ ХЕШУВАННЯ

З метою кількісного представлення часових характеристик операційних пристроїв хешування, скористаємось результатами синтезу комбінаційних схем, комутатора та конвеєрних регістрів для алгоритму SHA-1 на програмовану логічну інтегральну схему ALTERA EPF10K50-3: $T_{Pг}$=0.5 нс, T_{KC1}=10 нс, T_{KC81}=2.5 нс, $T_к$=2 нс. Скориставшись виразами (1–7), побудуємо графіки залежності такту роботи операційних пристроїв від їх структури (рис. 6), часу обробки повідомлень, розмір яких відповідає розміру інформаційних пакетів даних мережі Ethernet 46, 512, 1500 біт (рис.7).

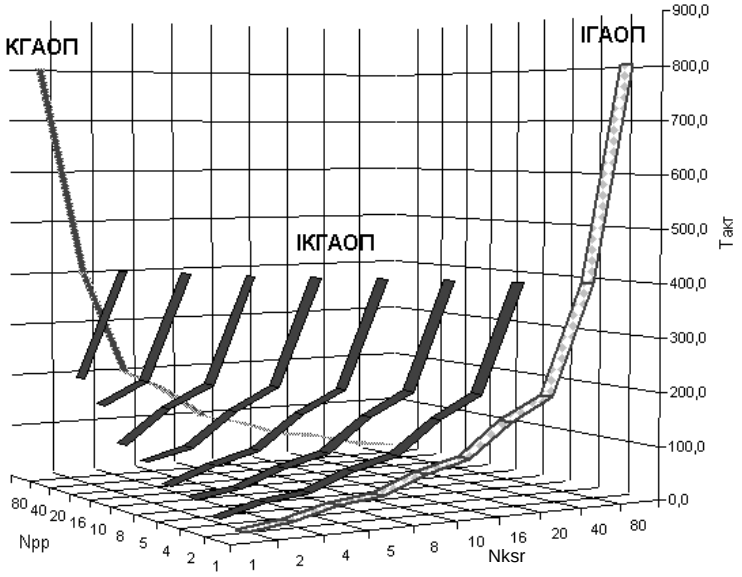

Рис. 7 – Графік залежності такту операційних пристроїв від їх структури.

Для ідентифікації структур операційних пристроїв на графіках введемо таке умовне позначення структур: *SN*(*Nksr*, *Npp*), де *SN* – код назви структури операційного пристрою ("і" – ІГАОП, "к" – КГАОП, "ік" – ІКГАОП), *Nksr*, *Npp* – параметри структур операційних пристроїв: кількість реалізованих комбінаційних схем і конвеєрних регістрів відповідно. Зауважимо, що розмір пакету на рис. 7 представлено у кількості буферів, необхідних для обробки інформаційних пакетів даних Ethernet. Аналіз графіків на рис. 6 і рис. 7 дозволяє встановити, що із збільшенням розміру даних, для яких необхідно проводити обчислення хеш-значення, конвеєрні та ітераційно-конвеєрні структури операційних пристроїв хешування дозволяють проводити обробку даних із меншими часовими затратами у порівнянні із ітераційними структурами.

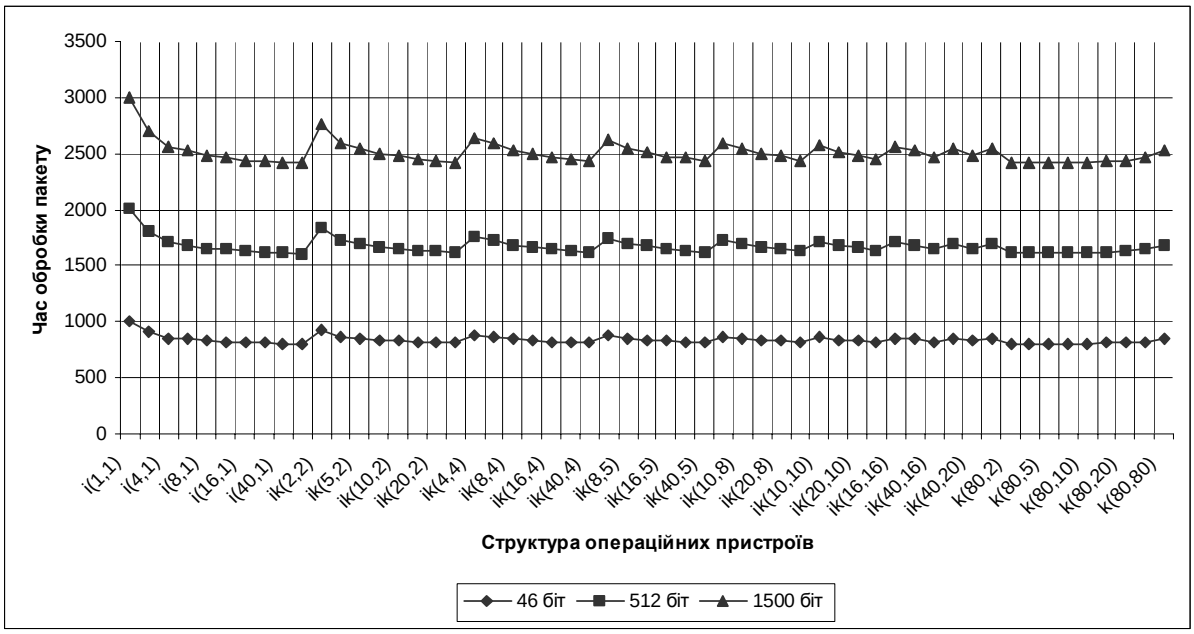

Рис. 7 – Графіки залежності часу обробки пакетів даних комп'ютерних мереж від структури операційних пристрої.

Однак, для галузей застосування, де розмір даних для обробки є в межах розміру одного буфера для обчислення розпису повідомлення, доцільно використовувати ітераційній структури
44 аційних пристроїв хешування.

4. ВИСНОВКИ

У роботі проведено дослідження структурної організації алгоритмів хешування MD5 і SHA-1, які використовуються для обробки даних у протоколі IPSec. Результати дослідження дозволили визначити базові структури операційних пристроїв для спеціалізованих процесорів хешування. До складу базових структур включено: ГАОП, КГАОП та ІКГАОП.

Для кожної з базових структур побудовано аналітичні формули, які пов'язують часові характеристики операційних пристроїв з параметрами їх структури. Використовуючи отримані аналітичні вирази та результати експериментального визначення часових параметрів складових операційних пристроїв при їх синтезі на програмовані логічні інтегральні схеми, отримано графіки залежностей часових характеристик операційних пристроїв від параметрів їх структури. Аналіз отриманих залежностей свідчить, що найменший такт роботи операційних пристроїв забезпечує конвеєрна структура.

Отримано графіки залежностей часу обробки заданої кількості буферів даних операційними пристроями різної структури, що дозволило встановити, що для обробки коротких повідомлень, які займають один буфер, доцільно використовувати ітераційні структури операційного пристрою. Для обробки більших за розміром повідомлень доцільно використовувати конвеєрні структури операційного пристрою.

5. НАПРЯМКИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

Отримані результати можна використати для оптимізації часових характеристик комплексних операційних пристроїв, які реалізують як хешування, так і симетричне блокове шифрування інформації.

Додатково, актуальною є задача вибору оптимального рівня представлення базових операцій алгоритмів хешування для забезпечення малого такту роботи операційних пристроїв при збереженні чи незначному збільшенні числа тактів для обробки повідомлень.

ЛІТЕРАТУРА

[1] S. Kent , R. Atkinson. Security Architecture for the Internet Protocol // Internet-Draft, May 1998.

[2] Столлингс. В. Криптография и защита сетей // М.: Вильямс. – 2001. – с. 481 –530.

[3] Коркішко Т. А. Багатоканальні апаратно-орієнтовані процесори симетричного блокового шифрування: Дис...канд.тех.наук: 05.13.05. – Львів., 2002. – 213 с.

[4] Rivest R. The MD5 Message-Digest Algorithm // RFC 1321, April 1992.

[5] Federal Information Proceedings Standard 180-1, Secure hash standard, National Institute of Standartization, US Department of Commerce, Washington D. C., April 1995.

[6] Preneel B., Rijmen V., and Bosselaers A. Recent developments in the design of conventional cryptographic algorithms / Computer Security and Industrial Cryptography – State of the Art and Evolution. – Springer-Verlag, 1998. – P. 123 – 145.

[7] Philips VMS747 Processor Application Notes. Philips Semiconductor. – 1999. – 30 p.

[8] Safe Net SafeXcel-2141 Processor Architecture. User Guide. – 2000. – 27 p.

[9] Net Octave NSP2000 Internet protocol Security Processor Datasheet. – 2001. – 15 p.

[10] HiFn Security Processors Selector Guide Hipp 7814, Hipp 7854, Hipp 7955. – 2001. – 12 p.

[11] Motorola MPC185TS/D, MPC184TS/D Communication Processors. Datasheet. – 2001. – 56 p.

[12] Мельник А.О. Спеціалізовані комп'ютерні системи реального часу. – Львів: Державний університет "Львівська політехніка", 1996. – 54 с.

[13] Мельник А.А. Процессоры обработки сигналов. – Львов, 1989. – 63 с. – (Препринт / АН УССР. Ин-т прикл. Проблем механики и математики; №29-89).

***Тимур Коркішко**, закінчив Державний університет "Львівська політехніка" за спеціальністю "Комп'ютерні та інтелектуальні системи і мережі" у 1997 році. У 2003 році отримав науковий ступінь кандидата технічних наук. З 2002 року працює старшим викладачем кафедри комп'ютерних наук Тернопільської академії народного господарства. Викладає курси: мови програмування, основи автоматизованого проектування засобів обчислювальної техніки. Області наукових інтересів: високошвидкісна криптографія, методологія розробки криптографічних процесорів і акселераторів, мови програмування. Автор більш як 25 наукових праць.*

***Леся Коркішко**, закінчила Державний університет "Львівська політехніка" за спеціальністю "Метрологія, стандартизація та сертифікація" у 1997 році. З 2002 року працює асистентом кафедри безпеки інформаційних технологій Тернопільської академії народного господарства. Викладає курси: основи захисту інформації, комплексне забезпечення комп'ютерної інформаційної безпеки. Області наукових інтересів: стандартизація і сертифікація засобів захисту інформації, методологія розробки криптографічних процесорів і акселераторів.*

***Руслан Шевчук**, з 1998 року студент інституту комп'ютерних інформаційних технологій Тернопільської академії народного господарства. З 2001 року працює інженером в спеціалізованій лабораторії програмного забезпечення кафедри комп'ютерних наук. Області наукових інтересів: процесори протоколу IPSec, криптографія, апаратна реалізація криптографічних алгоритмів.*