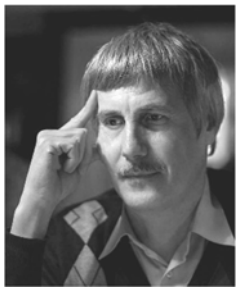
© Hilgurt S. Constructing optimal reconfigurable pattern matching tools for information security // Ukrainian Scientific Journal of Information Security, 2019, vol. 25, issue 2, pp. 74-81.

DOI: 10.18372/2225-5036.25.13824

МЕТОДИ ПОБУДОВИ ОПТИМАЛЬНИХ СХЕМ РОЗПІЗНАВАННЯ ДЛЯ РЕКОНФІГУРОВНИХ ЗАСОБІВ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ

Сергій Гільгурт

Інститут проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України



ГІЛЬГУРТ Сергій Якович, к.т.н., с.н.с.

Рік та місце народження: 1964 рік, Каракульський р-н Бухарської обл., Узбекистан.
Освіта: Київський інститут інженерів цивільної авіації (з 2000 року – Національний авіаційний університет), 1986 рік.
Посада: старший науковий співробітник ІПМЕ ім. Г.Є. Пухова НАН України з 2004 року.
Наукові інтереси: реконфігуровні обчислення, технічний захист інформації.
Публікації: 115 наукових публікацій, серед яких авторські свідоцтва на винаходи, наукові статті, матеріали та тези доповідей на конференціях.
E-mail: hilgurt@nau.edu.ua.
Orcid ID: 0000-0003-1647-1790.

Анотація. *Через сталий зріст об'єму мережевого трафіку, кількості та складності атак програмні рішення вже не встигають в реальному часі розпізнавати сигнатури для таких засобів технічного захисту, як мережеві системи виявлення вторгнень, антивірусні сканери, фільтри протидії мережевим хробакам, тощо. Тому розробники все частіше звертають увагу на реконфігуровні (на базі ПЛІС) апаратні рішення, що поєднують в собі продуктивність спецпроцесорів із гнучкістю майже як у програмного забезпечення. На сьогоднішній день відомі декілька підходів до побудови сигнатурних засобів інформаційного захисту з використання програмованої логіки. Але жоден з них не демонструє явних переваг перед іншими. У даній статті запропоновані методи підвищення ефективності реконфігуровних засобів технічного захисту шляхом синтезу оптимальних схем розпізнавання, які найкращим чином використовують переваги кожного з підходів та окремих технічних рішень.*

Ключові слова: *захист інформації, ПЛІС, сигнатура, комбінування підходів, ефективність, оптимізація.*

Вступ

Проблеми безпеки комп'ютерних систем і мереж, на жаль, с часом стають лише актуальніше. Такі засоби технічного захисту інформації, як системи виявлення вторгнень [1], антивірусні сканери та фільтри проти мережевих хробаків, робота яких заснована на сигнатурному аналізі інтенсивного потоку даних, вирішують в реальному часі обчислювально складну задачу множинного розпізнавання рядків [2, 3]. У зв'язку зі збільшенням числа і складності вторгнень, а також через припинення зростання частоти мікропроцесорів програмні рішення вже не впорюються з підвищеними вимогами щодо швидкодії. З цих причин розробники систем технічного захисту все більше уваги приділяють реконфігуровним засобам на базі програмованих логічних інтегральних схем (ПЛІС) [4, 5]. Висока продуктивність програмованої логіки в поєднанні з гнучкістю, близькою до програмної, якнайкраще відображає динамічну природу галузі інформаційного захисту. В якості платформи її практичного застосування найбільш придатними виявилися реконфігуровні обчислювачі (або прискорювачі). Такі пристрої зазвичай містять одну мікросхему ПЛІС типу Field Programmable Gate Array (FPGA), бортову пам'ять, канали швидкісного обміну інформацією, інтерфейс для підключення до традиційних комп'ютерів та інші компоненти [6].

Застосування реконфігуровних обчислювачів дозволило дещо покращити ситуацію з сигнатурними системами інформаційної безпеки. Але збільшення об'ємів мережевого трафіку триває, як і зростання кількості та складності зовнішніх атак на локальні мережі. Тому завдання ефективного використання засобів захисту інформації на базі ПЛІС лишається *актуальною науково-технічною проблемою.*

Метою даної роботи є розробка методів підвищення ефективності реконфігуровних засобів захисту інформації на основі аналізу світового досвіду їх побудови.

Мережеві системи виявлення вторгнень на базі ПЛІС

Історично першими і, як наслідок, найбільш дослідженими сигнатурними засобами інформаційного захисту, при побудові яких почали використовувати ПЛІС типу FPGA, стали системи виявлення вторгнень (СВВ) [7]. Тому, не втрачаючи загальності міркування, розглянемо типові функції реконфігурованих засобів захисту та показники їх ефективності саме на прикладі СВВ.

В залежності від об'єкту, що захищається, розрізняють СВВ, які [8]:

– контролюють окремі комп'ютери;

– аналізують пакети мережевого трафіку всієї локальної мережі.

Найбільший ефект від застосування апаратних рішень досягається для засобів другого типу – мережевих систем виявлення вторгнень (МСВВ), відповідний англомовний термін - Network Intrusion Detection System (NIDS). Тому в подальшому розглядаються саме такі системи.

Найважливішим компонентом МСВВ є модуль розпізнавання (МР), від характеристик якого значною мірою залежать показники ефективності системи в цілому. Даний модуль вирішує обчислювально складну задачу множинного розпізнавання рядків, тобто перевіряє зміст мережевих пакетів на наявність певних послідовностей символів – так званих патернів, які входять до складу сигнатур – описів відомих атак. Бази даних сигнатур сучасних засобів технічного захисту містять від десятків тисяч до мільйонів патернів, які потрібно одночасно відшукувати у вхідному потоці даних в реальному масштабі часу.

Основні показники ефективності реконфігурованих МСВВ як типових сигнатурних засобів захисту інформації можна поділити на:

– вартісні;

– швидкісні або показники продуктивності;

– функціональні.

До вартісних показників належать: обсяг логічних ресурсів програмованої логіки, задіяної для створення цифрової схеми, витрати на пам'ять (як зовнішню відносно кристалу ПЛІС, так і внутрішню – блочну пам'ять BRAM та розподілену у вигляді тригерів логічних комірок), а також інші витрати, що складають загальну вартість володіння, включи розробку, виготовлення, програмування та експлуатацію системи.

До параметрів продуктивності відносять об’єм словнику (тобто кількість патернів, які розпізнає система), швидкодію засобу, яка характеризується або часом затримки розповсюдження даних від входу до виходу або пропускною здатністю, а також передбачуваність пропускної здатності.

Важливим проміжним показником, який пов'язує швидкісні характеристики з вартісними, є масштабованість – здатність нарощувати характеристики продуктивності без завеликих додаткових ресурсних витрат. Розрізняють масштабованість за пропускною здатністю, за об’ємом словнику патернів та за довжиною патернів.

До функціональних показників відносяться, наприклад, спроможність МСВВ працювати у режимі запобігання вторгнень, здатність до динамічного оновлення словнику патернів без припинення процесу розпізнавання, можливість протидіяти цілеспрямованим атакам на систему захисту тощо.

Порівняння найбільш поширених підходів до побудови реконфігуровних модулів розпізнавання

В результаті аналізу досвіду чисельних розробок дослідників зі всього світу виявляється, що при створенні МСВВ найкращі здібності продемонстрували три підходи, які використовують наступні технології та відповідні технічні рішення, що на них засновані:

– цифрові компаратори (ЦК) – асоціативна пам'ять (АП);

– хеш-функції (ХФ) – фільтр Блума (ФБ);

– скінченні автомати (СА) – алгоритм Ахо-Корасік (АК).

Детальне вивчення особливостей кожного з підходів, яке було проведено відповідно в трьох нещодавно опублікованих роботах автора [9 – 11], дає змогу порівняти їх технічні можливості щодо показників ефективності. В табл. 1 наведені результати порівняльного аналізу підходів.

Таблиця 1

Результати порівняння основних підходів до побудови реконфігуровних МСВВ

№	Показник		Підхід		
			Асоціативна пам'ять	Фільтр Блума	Алг. Ахо-Корасік
1.	Витрати на логіку		---	+	+++
2.	Витрати пам'яті	розподіленої	---	+	+++
3.		блочної	+++	+	---
4.		зовнішньої	+++	+++	---
5.	Швидкодія		+++	+	-
6.	Передбачуваність пропускної здатності		+++	---	+++
7.	Масштабованість	за пропускною здатністю	+++	+	-
8.		за об’ємом словнику	---	+++	---
9.		за довжиною патернів	-	+++	+++
10.	Використання надлишковості		+	---	+++
11.	Функціональні показники	динамічне оновлення	---	+	+++
12.		протидія атакам на МСВВ	+++	---	+
13.		режим запобігання вторгнень	+++	+	-
14.	Суттєвий недолік, який зводить нанівець головні переваги підходу		завелике споживання ресурсів	фіксована довжина патернів	"вибуховий" зріст об’єму пам'яті

Позначення: "**+++**" – суттєва перевага; "**+**" – помірна перевага; "**---**" – суттєвий недолік; "**-**" – помірний недолік;

Як бачимо, жоден з досліджених підходів не демонструє явних переваг перед іншими та не відповідає повністю вимогам, що пред'являються до МСВВ, кожен має власні позитивні риси та недоліки.

Так, наприклад, ЦК та побудовані на їх основі різновиди АП забезпечують максимальну швидкодію, але витратніше інших підходів за споживанням апаратних ресурсів та електроенергії, вони також програють в плані масштабування. Фільтр Блума більш економічний та краще масштабується, але накладає обмеження на довжину патернів; він також вимагає додаткових витрат на доуточнення здобутих результатів через системну наявність помилок розпізнавання другого роду. Скінченні автомати більш економні щодо витрат на логіку, забезпечують стабільну, але відносно невисоку пропускну здатність, складні в побудові, призводять до "вибухового" зростання об'єму пам'яті для великих словників сигнатур.

Відсутність лідируючого напрямку, який би перевершував конкурентні рішення за всіма показниками, призводить до того, що розробники, по-перше, пропонують чисельні модифікації основних підходів, намагаючись позбутися їх недоліків, по-друге – комбінують підходи в різноманітних поєднаннях. Але ці спроби носять евристичний, несистемний характер. Відчувається брак формалізації та узагальнення, що не дозволяє перевести проблему з інженерного на науковий рівень.

Тому на основі вивчення та систематизації існуючого досвіду побудови реконфігуровних МСВВ були запропоновані методи, які дозволяють формалізувати ідею сумісного використання різних підходів, а також за рахунок оптимізації максимізувати ефективність використання переваг кожного з них.

Метод паралельного комбінування (МПрКм)

Суть методу полягає в поділі (кластеризації) набору патернів, які повинні відшукуватися модулем розпізнавання, на підгрупи (кластери), а також у синтезі в складі МР такої ж кількості різнорідних блоків розпізнавання (БР), кожен з яких найбільш результативно відшукує патерни відповідної підгрупи, максимально використовуючи при цьому переваги задіяного в ньому підходу. Найвища ефективність при цьому досягається за рахунок охоплення обох процесів – кластеризації та вибору БР – загальною процедурою оптимізації.

Підґрунтям методу є той факт, що патерни, які входять до словнику сигнатур, розрізняються за довжиною та за властивістю самоподоби, в наслідок чого ефективність, з якою вони опрацьовуються, також відмінна та залежить від підходу до побудови засобу розпізнавання. Іншим чинником на користь даного методу є фіксований набір ресурсів реконфігуровних обчислювачів, на базі яких будуються МСВВ. Використання лише одного з можливих способів розпізнавання призводить до того, що частина ресурсів (наприклад, логічних) використовується майже повністю, а інші (ресурси блочної чи бортової пам'яті) – не задіяна взагалі. Як наслідок, максимально можлива ефективність не досягається.

На рис. 1. подано схематичне зображення структури МПрКм.

На етапі синтезу набір патернів P, що підлягає розпізнаванню, за допомогою процедури кластеризації розбивається на n підгруп P_i, $i = 1 \dots n$. Після синтезу *в процесі функціонування* кожний блок розпізнавання БР$_i$ в складі МР здійснює пошук сигнатур відповідної підгрупи у потоці вхідних даних, які подаються одночасно на входи всіх БР. В разі виявлення збігу фрагменту вхідних даних з якимось патерном активується потрібний сигнал розпізнавання.

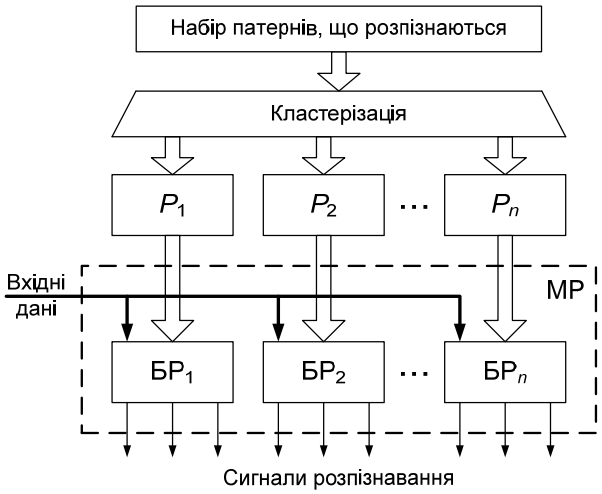


Рис. 1. Схематичне подання структури методу паралельного комбінування

Для досягнення максимальної ефективності процес формування структури МР відбувається під управлінням загальної процедури оптимізації.

Змінними параметрами при цьому є кількість підгруп патернів n та їх склад P_i, а також варіанти реалізації кожного з блоків БР$_i$, які вибираються з бібліотеки готових компонентів.

Критерії оптимізації можуть використовуватися різними в залежності от потреб користувачів МСВВ. В довільному випадку метою оптимізації є мінімізація або максимізація певної цільової функції, в якості якої виступає чисельне значення певного технічного параметру: споживані ресурси, швидкодія, продуктивність, тощо.

Алгоритм, що реалізує МПрКм, варіює змінні параметри, обчислюючи на кожному кроці ресурсні та часові характеристики кожного з БР та МР в цілому.

Безпосередньо ресурсні та часові характеристики реконфігуровного модулю розпізнавання можна знайти шляхом синтезу його цифрової схеми за допомогою інструментальних засобів роботи з ПЛІС. Але цей процес потребує забагато часу [12], що унеможливлює його використання при реалізації МПрКм.

Тому було запропоновано методику прискореного обчислення характеристик БР та МР. Її суть полягає у створенні для кожного бібліотечного компонента так званої функції-калькулятора. Така функція, маючи на вході заданий набір патернів підгрупи P_i, повинна обчислювати об'єм ресурсів, який вживатиме i-й БР, що буде розпізнавати цю підгрупу патернів, а також чисельне значення часової затримки, яку він матиме в результаті синтезу.

У зв'язку з відмінностями різних підходів до розпізнавання, функції-калькулятори мають будуватися в різний спосіб. Наприклад, для блоків розпізнавання, що будуються на базі АП, схеми якої мають прозору та регулярну структуру, функція-калькулятор може бути сформована шляхом прямого підрахунку потрібних ресурсів та часових затримок.

Розглянемо як приклад складання ресурсної складової функції-калькулятору для базової схеми розпізнавання патернів на цифрових компараторах BSCAM (див. рис. 1, а у роботі [9]).

В загальному випадку ресурси, що потрібні для синтезу в ПЛІС *i*-го БР можуть бути обчислені в умовних одиницях, еквівалентних в сенсі витрат пошуковим таблицям LUT:

$$R_i = L_i + \alpha F_i + \beta B_i + \gamma M_i, \quad (1)$$

де L_i – ресурси логіки ПЛІС, які потрібні для синтезу *i*-го блоку (кількість пошукових таблиць LUT); F_i – ресурси розподіленої пам'яті ПЛІС (кількість тригерів), B_i – ресурси блочної пам'яті ПЛІС (Мб), M_i – ресурси зовнішньої пам'яті – бортової пам'яті реконфігурованого обчислювача (Мб), α, β, γ – коефіцієнти нормалізації ресурсів різного типу відносно ресурсів логіки (пошукових таблиць LUT).

Оскільки в логічних комірках більшості сучасних ПЛІС кількість пошукових таблиць LUT та тригерів однакова (α = 1), а також завдяки тому, що схеми розпізнавання на АП не потребують блочної та зовнішньої пам'яті, вираз (1) для базової схеми цифрових компараторах BSCAM спрощується до

$$R_{\text{BSCAM}} = L_{BSCAM} + F_{BSCAM}, \quad (2)$$

де L_{BSCAM} і F_{BSCAM} – відповідно кількість LUT і тригерів в схемі.

Пошукові таблиці LUT в схемі BSCAM використовуються, по-перше, для синтезу компараторів CMP, по-друге – для створення багатовходової схеми "I", що будується у вигляді конвеєра [9]:

$$L_{BSCAM} = L_{CMP} + L_{\&}. \quad (3)$$

Особливістю схеми BSCAM є той факт, що вона потребує стільки компараторів, скільки символів сумарно міститься у всіх патернах набору, що має розпізнаватися:

$$\Omega = \sum_{j=m_{\min}}^{m_{\max}} \delta_j \cdot j, \quad (4)$$

де j – номер патерну в наборі, $m_{\min}$ – довжина найкоротшого патерну, $m_{\max}$ – довжина найдовшого патерну, δ_j – функція розподілу довжин патернів.

Один цифровий компаратор CMP, що розпізнає символ у байтовому кодуванні, при побудові схем АП потребує дві пошукові таблиці LUT на 4 чи на 6 входів, або одну LUT, якщо вона 8-входова. Введемо функцію-кваліфікатор

$$\Lambda(x) = \begin{cases} 1, & x = 8 \\ 2, & x < 8 \end{cases},$$

де x – кількість входів пошукової таблиці LUT для заданої ПЛІС. Тоді кількість LUT, потрібна для створення всіх компараторів схеми BSCAM дорівнюватиме

$$L_{CMP} = \Lambda(x) \cdot \Omega. \quad (5)$$

Кількість LUT, потрібних для об'єднання *j* входів каскадною схемою "I", з урахуванням того факту, що на *x*-входовій LUT можна синтезувати логічну схему "I" не більш, чим на *x* входів, дорівнює для кожного патерну

$$L_j = L(j) = \left\lceil \frac{j-1}{x-1} \right\rceil. \quad (6)$$

Кількість LUT для складання всіх конвеєрів всіх патернів підраховується аналогічно (4) з урахуванням (6):

$$L_{\&} = \sum_{j=m_{\min}}^{m_{\max}} \delta_j \cdot \left\lceil \frac{j-1}{x-1} \right\rceil. \quad (7)$$

Кількість тригерів, потрібних для створення схеми BSCAM, складається з їх кількості, потрібної для побудови вхідного конвеєра F_{RG}, кількості тригерів в конвеєрі розгалуження для підвищення здатності навантаження (fan-out) виходів регістрів вхідного конвеєра $F_{\text{fan-out}}$ та кількості тригерів в конвеєрі для об'єднання по "I" виходів всіх компараторів для всіх патернів $F_{\&}$:

$$F_{BSCAM} = F_{RG} + F_{fan-out} + F_{\&}. \quad (8)$$

Вхідний конвеєр, по якому просувається потік символів, що розпізнаються, має бути не коротшим за довжину найдовшого патерну всієї підгрупи $m_{\max}$ та шириною в один байт. Тому кількість тригерів, потрібних для його побудови:

$$F_{RG} = 8 \cdot m_{\max}. \quad (9)$$

Пошук числа $F_{\text{fan-out}}$ виявляється дещо складнішим порівняно з попередніми розрахунками. Навантаження на вихідні ланцюги регістрів вхідного конвеєру нерівномірне. Якщо з його перших $m_{\min}$ ступенів сигнали використовуються для розпізнавання всіх патернів підгрупи, то з останнього – тільки для розпізнавання найдовших патернів. Виконуючи всі потрібні розрахунки, отримуємо:

$$F_{fan-out} = m_{\min}\left\lceil \frac{\sigma-1}{y-1} \right\rceil + \sum_{i=m_{\min}+1}^{m_{\max}} \left\lceil \frac{\sum_{i=j}^{m_{\max}} \delta_i - 1}{y-1} \right\rceil, \quad (10)$$

де σ – кількість патернів у групі, *y* – здатність навантаження виходів тригерів для заданої ПЛІС.

Кількість тригерів в конвеєрі для об'єднання по "I" для кожного патерну відрізняється на одиницю від кількості LUT у цьому ж конвеєру $L_{\&}$ згідно (7), тому що після останнього каскаду схеми "I" тригер не потрібен:

$$F_{\&} = L_{\&} - 1 = \sum_{j=m_{\min}}^{m_{\max}} \delta_j \cdot \left\lceil \frac{j-1}{x-1} \right\rceil - 1. \quad (11)$$

Підставляючи (5) та (7) в (3), а також (9), (10) та (11) в (8), а потім – (3) та (8) у (2), та враховуючи той факт, що тригери та пошукові таблиці LUT, які задіяні при синтезі конвеєрної схеми багатовходового логічного елементу "І", можуть бути використані сумісно з однієї логічної комірки ПЛІС, отримуємо загальну кількість обчислювальних ресурсів у базовій схемі BSCAM для блоку розпізнавання на базі ЦК:

$$R_{BSCAM} = \sum_{j=m_{\min}}^{m_{\max}} \delta_j \left(\Lambda(x) j + \left\lceil \frac{j-1}{x-1} \right\rceil \right) + $$

$$+ 8 m_{\max} + m_{\min} \left\lceil \frac{\sigma - 1}{y-1} \right\rceil + \sum_{i=m_{\min}+1}^{m_{\max}} \left\lceil \frac{\sum_{i=j}^{m_{\max}} \delta_i - 1}{y-1} \right\rceil . \quad (12)$$

Проаналізуємо здобутий результат. Функція-калькулятор ресурсів (12) для блоку розпізнавання, побудованому з використанням базової схеми BSCAM на цифрових компараторах, залежить, з одного боку, від параметрів множини патернів підгрупи, що розпізнається: σ, $m_{\min}$, $m_{\max}$, δ, з іншого – від властивостей мікросхеми ПЛІС, що використана в реконфігуровному обчислювачі: x, y. В процесі виконання процедури оптимізації параметри підгрупи патернів є змінними, в той час як характеристики ПЛІС – константами.

Метод послідовного каскадування (МПсКс)

Розглянутий вище метод паралельного комбінування використовує той факт, що патерни, які входять до словнику сигнатур, суттєво розрізняються між собою, зокрема, за довжиною. Оптимізаційна процедура розподіляє їх по блоках розпізнавання, найбільш придатних для кожного різновиду патернів для подальшого паралельного розпізнавання. Але при цьому кожен з патернів має бути розпізнаним повністю, щоб відповідний сигнал тривоги був активований.

Дійсно, для виявлення факту збігу фрагменту вхідних даних з якимось патерном потрібно, щоб співпали одночасно всі символи патерну, для чого, фактично, виконується логічна операція "І" на таке число змінних, що дорівнює довжині шуканого підрядку. З іншого боку, щоб зробити висновок, що поточний фрагмент даних не співпадає з патерном, достатньо виявити всього одну розбіжність, і тоді порівняння решти байтів стає зайвим.

Метод МПсКс використовує цей факт для підвищення ефективності процесу розпізнавання. Його суть полягає у розбитті операції порівняння на послідовні етапи, на кожному з котрих здійснюється часткове розпізнавання відповідного фрагменту довгого патерну, але тільки у випадку виявлення збігу на попередньому етапі. В структурі МР для кожного з часткових розпізнавань створюється окремий каскад КС$_i$, $i = 1 \ldots n$, де n – кількість фрагментів розбиття (кількість каскадів).

На рис. 2. подано схематичне зображення структури МПсКс. На етапі синтезу кожний патерн всередині вхідної підгрупи P, ділиться на n фрагментів, в результаті чого ця підгрупа також ділиться на n частин Q_i, $i = 1 \ldots n$. Після синтезу в процесі функціонування кожний каскад КС$_i$ в складі МР виконує розпізнавання відповідних фрагментів патернів, але лише у випадку виявлення збігу для відповідного фрагменту з попереднього каскаду. В разі виявлення збігу для якогось фрагменту в останньому каскаді КС$_n$, активується потрібний сигнал розпізнавання.

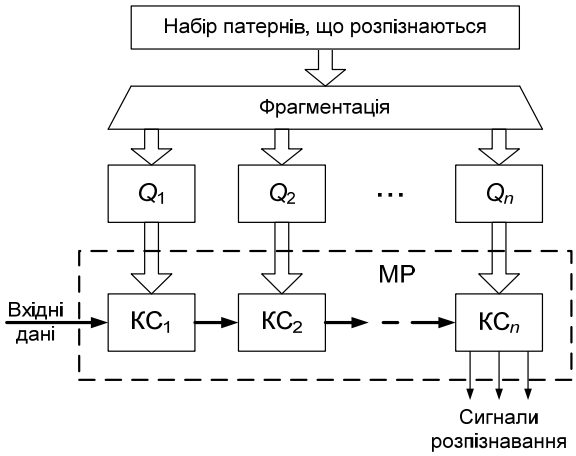


Рис. 2. Структура модулю розпізнавання за методом послідовного каскадування

Виходячи з припущення, що збіг вхідних даних з більш довгим фрагментом патерну є менш вірогідною подією, можна очікувати, що потреба в використанні кожного наступного каскаду буде виникати рідше порівняно з попереднім, внаслідок чого вимоги до швидкодії (та, як наслідок – споживані ресурси) каскадів знижатимуться за зростанням номеру каскаду i аж до можливості реалізації останніх каскадів програмними засобами. Водночас швидкодія каскаду може бути досягнута тим вища, чим коротше шукані в ньому підрядки. В результаті побудова МР за каскадною схемою може призвести одночасно як до підвищення швидкодії МСВВ, так і до скорочення потрібних ресурсів.

Зазначимо, що недоліком методу МПсКс є погіршення такого функціонального показника як здатність протидії цілеспрямованим атакам на засоби захисту інформації. Прикладом подібної дії може бути атака алгоритмічної складності – навмисне насичення порушником мережевого трафіку хибними пакетами спеціального змісту, розпізнавання яких призводить до лавиноподібного зростання числа обчислювальних операцій, здатне унеможливити коректну роботу МСВВ [13]. Цей факт треба враховувати при застосуванні даного методу при побудові засобів захисту.

Вдале розбиття патернів на фрагменти для методу МПсКс не є тривіальною задачею.

По-перше, з вищенаведеного опису зрозуміло, що фрагменти патернів для перших каскадів повинні бути коротшими, а для останніх – більш довгими. з іншого боку, велика кількість каскадів призводить до зростання складності схеми та накладних витрат.

По-друге, виникає протиріччя: з одного боку, схеми розпізнавання більш коротких фрагментів

швидші та простіші, с другого – для таких фрагментів вище вірогідність хибного розпізнавання, тобто наступні каскади будуть частіше спрацьовувати дарма. У граничному випадку, коли перший фрагмент скорочений до одного символу, другий каскад у середньому буде запускатися на кожному 256-му байти вхідної послідовності для кожного патерну.

По-третє, під час розбиття патернів на фрагменти можна взяти до уваги властивість самоподоби. Тобто між патернами підгрупи доцільно виявляти однакові фрагменти, які можуть спільно використовувати одні й ті самі розпізнавальні ресурси, знижуючи тим самим апаратні витрати.

Для знаходження найкращого значення кількості фрагментів n та максимально ефективного розбиття патернів підгрупи P на n фрагментів також пропонується використовувати процедуру оптимізації. Критерії можуть бути такими ж, як і для методу МПрКм. Змінними параметрами є кількість фрагментів n, співвідношення довжин фрагментів і варіанти реалізації кожного з каскадів $КС_i$, які також вибираються з бібліотеки готових компонентів.

Особливістю методу МПсКс є необхідність крім фіксованих властивості питомій підгрупи патернів враховувати статистичні параметри вхідного інформаційного потоку, зокрема, для МСВВ – імовірнісні характеристики мережевого трафіку.

Оскільки МПсКс більш ефективний для довгих патернів, його доцільно застосовувати для розпізнавання патернів не зі всього словнику сигнатур, а лише для окремій його частки, тобто для певній підгрупи патернів.

Метод вертикального об’єднання (МВрОб)

Суть методу полягає в сполучанні кількох підходів (або технік) в одному модулі таким чином, що жоден з них не може бути відокремленим від інших. Найчастіше при такому об'єднанні один з підходів застосовується для вирішення якоїсь допоміжної функції в складі структури, що функціонує за принципами іншого підходу. Наприклад, в роботі [14] для підвищення швидкодії скінченного автомату вхідні дані поділяються на блоки по декілька байтів, що обробляються як окремі символи, при цьому такі блоки розпізнається фільтрами Блума, а схема в цілому діє за алгоритмом Ахо-Корасік. Об'єднуватись можуть не тільки підходи а й технології, на яких підходи базуються. Наприклад розробка [15] схожа на вищезгадану, але замість ФБ в неї використовуються окремі хеш-функції в поєднанні з АП, тобто, об’єднуються два повноцінних підходи та технологічна основа третього. В роботі [16] успішно поєднаються технологія хеш-функцій та підхід на основі АП.

При використанні даного методу набор патернів, що розпізнаються, ніяким чином не діліться на підгрупи (рис. 3). Тобто побудований за методом МВрОб модуль діє як єдиний функціональний блок.

Слід зауважити, що при використанні МВрОб можуть об'єднуватись ні тільки різні підходи, але й різні модифікації одного підходу. Наприклад, техніка bit-split [17] виявляється ортогональною до інших різновидів алгоритму АК, і в разі застосування спільно з основним рішенням, дозволяє зменшити споживані ресурси на певний відсоток.

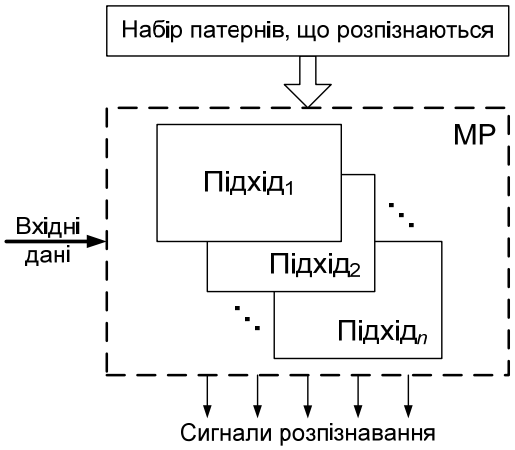


Рис. 3. Структура модулю розпізнавання за методом вертикального об’єднання.

На відміну від МПрКм та МПсКс метод МВрОб, точніше, принципи, закладені в нього, досить активно використовуються розробниками реконфігуровних засобів технічного захисту інформації. Внесок автора крім формалізації та системного опису цих принципів у вигляді методу, полягає також в створенні алгоритму його реалізації з використанням оптимізаційних процедур. Згідно цього алгоритму для низки технічних рішень (якими можуть бути відомі підходи, технології, техніки та конкретні схеми) створюється багатовимірна таблиця сумісності (ТС), яка для кожної можливої комбінації рішень задає функцію-об'єднувач. Така функція визначає, по-перше, чи можуть її вхідні аргументи функціонувати спільно (тобто чи існує схема, згідно якої відповідні рішення можуть об'єднатися в модуль розпізнавання методом МВрОб), по-друге – в якому порядку у випадку такої можливості відбудовуватиметься така взаємодія, потретє – на який коефіцієнт покращуватимуться показники ефективності провідного рішення підпорядкованими в результаті об'єднання. Використання ТС разом з функціями-калькуляторами дозволяє охопити процес створення МР за методом МВрОб циклом оптимізації з метою максимізації обраної цільової функції для заданої підгрупи патернів. Змінними параметрами при цьому є технічні рішення – аргументи функції-об'єднувача таблиці сумісності.

Висновки

На сьогоднішній день у світі накопичений величезний досвід зі створення реконфігуровних засобів захисту інформації, в тому числі – з побудови мережевих систем виявлення вторгнень. Найбільш відомими підходами до їх побудови є: 1) цифрові компаратори (ЦК) та асоціативна пам'ять (АП); 2) хеш-функції (ХФ) – фільтр Блума (ФБ); 3) скінченні автомати (СА) – алгоритм Ахо-Корасік (АК). При цьому жоден з відомих підходів не має значної переваги перед іншими за досягнутими технічними показниками. Розробниками запропоновано також безліч покращень, модифікацій та комбінованих схем. Систематизація наявних знань та науковий підхід до проблеми дали можливість розробити ряд методів підвищення ефективності МСВВ та інших сигнатурних засобів захисту

на базі ПЛІС шляхом синтезу оптимальних структур розпізнавання, які найкращим чином використовують переваги кожного з підходів та окремих технічних рішень.

Наступним кроком даного дослідження планується випробування здобутих теоретичних результатів з використанням обчислювальних експериментів.

Література

[1]. С. Казмірчук, А. Корченко, Т.І. Паращук, "Аналіз систем виявлення вторгнень", *Захист інформації,* Т. 20, № 4, С. 259-276, 2018.

[2]. Б. Смит, *Методы и алгоритмы вычислений на строках. Теоретические основы регулярных вычислений. Пер. с англ.* М.: Вильямс, 2006, 496 с.

[3]. С. Гільгурт, "Множинне розпізнавання рядків у системах виявлення вторгнення на базі реконфігуровних обчислювачів", *Сучасні комп'ютерні системи та мережі: розробка та використання: матеріали 5-ої Міжнар. наук.-техн. конф. ACSN-2011*, 29 вересня – 01 жовтня 2011, Львів, Україна. Л.: Вид-во Нац. ун-ту «Львів. політехніка», С. 54–56, 2011.

[4]. V. Paxson, K. Asanovic, S. Dharmapurikar, J. Lockwood, R. Pang, R. Sommer, N. Weaver, "Rethinking Hardware Support for Network Analysis and Intrusion Prevention", *Proceedings of the First USENIX Workshop on Hot Topics in Security*, pp. 63-68, 2006.

[5]. H. Chen, Y. Chen, D.H. Summerville, "A Survey on the Application of FPGAs for Network Infrastructure Security", *IEEE Communications Surveys and Tutorials,* pp. 541-561, 2011.

[6]. С. Гильгурт, "Реконфигурируемые вычислители. Аналитический обзор", *Электронное моделирование*, Т. 35, № 4, С. 49-72, 2013.

[7]. Ю. Коростиль, С. Гильгурт, "Принципы построения сетевых систем обнаружения вторжений на базе ПЛИС", *Моделювання та інформаційні технології. Зб. наук. пр. ІПМЕ ім. Г.Є. Пухова НАН України*, Вип. 57, С. 87-94, 2010.

[8]. А. Лукацкий, *Обнаружение атак*, СПб.: БХВ-Петербург, 2001, 624 с.

[9]. С. Гільгурт, "Побудова асоціативної пам'яті на цифрових компараторах реконфігуровними засобами для вирішення задач інформаційної безпеки", *Электронное моделирование*, Т. 41, № 3, С. 59-80, 2019.

[10]. С. Гільгурт, "Побудова фільтрів Блума реконфігуровними засобами для вирішення задач інформаційної безпеки", *Безпека інформації*, Т. 35, № 1, С. 53-58, 2019.

[11]. С. Гільгурт, "Побудова скінченних автоматів реконфігуровними засобами для вирішення задач інформаційної безпеки", *Захист інформації*, Т. 21, № 2, С. 111-120, 2019.

[12]. В. Евдокимов, А. Давиденко, С. Гильгурт, "Организация централизованной генерации файлов конфигураций для аппаратных ускорителей задач информационной безопасности", *Моделювання та інформаційні технології*, № 81, С. 3-11, 2017.

[13]. С. Гильгурт, Б. Дурняк, Ю. Коростиль, "Противодействие атакам алгоритмической сложности на системы обнаружения вторжений", *Моделювання та інформаційні технології*, № 71, С. 3-12, 2014.

[14]. S. Dharmapurikar, J.W. Lockwood, "Fast and scalable pattern matching for network intrusion detection systems", *IEEE Journal on Selected Areas in Communications*, Article Vol. 24, no. 10, pp. 1781-1792, Oct 2006.

[15]. Y.H. Cho, W.H. Mangione-Smith, "A pattern matching co-processor for network security", in 42nd *Design Automation Conference*, Anaheim, CA, Jun 13-17 2005, LOS ALAMITOS: IEEE Computer Soc, in Design Automation Conference DAC, 2005, pp. 234-239.

[16]. I. Sourdis, D. Pnevmatikatos, S. Vassiliadis, "Scalable Multigigabit Pattern Matching for Packet Inspection", *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Vol. 16, no. 2, pp. 156-166, 2008.

[17]. L. Tan, T. Sherwood, "A high throughput string matching architecture for intrusion detection and prevention", *in 32nd International Symposium on Computer Architecture*, Madison, WI, Jun 04-08 2005, LOS ALAMITOS: IEEE Computer Soc, in Conference Proceedings Annual International Symposium on Computer Architecture, 2005, pp. 112-122.

УДК 004.274:004.056

Гильгурт С. Методы построения оптимальных схем распознавания для реконфигурируемых средств информационной безопасности

Аннотация. *В связи с постоянным ростом объема сетевого трафика, количества и сложности атак программные решения уже не успевают в реальном времени распознавать сигнатуры для таких средств технической защиты, как сетевые системы обнаружения вторжений, антивирусные сканеры, фильтры противодействия сетевым червям и т.п. Поэтому разработчики все чаще обращают внимание на реконфигурируемые (на базе ПЛИС) аппаратные решения, совмещающие производительность спецпроцессоров с гибкостью как у программного обеспечения. На сегодняшний день известны несколько подходов к построению сигнатурных средств информационной защиты с использованием программируемой логики. Но ни один из них не демонстрирует явных преимуществ перед другими. В данной статье предложены методы повышения эффективности реконфигурируемых средств технической защиты посредством синтеза оптимальных схем распознавания, которые наилучшим образом используют преимущества каждого из подходов и отдельных технических решений.*

Ключевые слова: *защита информации, ПЛИС, сигнатура, комбинирование подходов, эффективность, оптимизация.*

Hilgurt S. Constructing optimal reconfigurable pattern matching tools for information security

Abstract. *Signature-based security tools such as network intrusion detection systems, anti-virus scanners, filters against network worms and other similar systems perform in real time computation-intensive task of multi-pattern string matching against tens of thousands or even millions of predefined malicious patterns. Due to rising traffic rates, increasing number and sophistication of attacks and the collapse of Moore's law for sequential processing, traditional software solutions can no longer meet the high requirements of today's security challenges. Therefore, designers pay more attention to hardware approaches to accelerate pattern matching. The reconfigurable devices based on Field Programmable Gate Arrays (FPGA) combining the flexibility of software and the near-ASIC performance, have become increasingly popular for this purpose. The state-of-the-art solutions made in this area around the world were analyzed. There are three main approaches to fulfill the pattern matching using FPGA. The techniques (and underlying technologies) of these approaches are: content addressable memory (based on digital comparators), Bloom filter (based on hash-functions) and Aho-Corasick algorithm (based on finite automata). But none of them shows clear advantages over others. In this article, we propose a set of methods to increase the effectiveness of reconfigurable security tools by synthesizing optimal recognition modules that maximize the benefits of each approach. The Parallel Combination Method divides a set of patterns between several matching blocks that use different approaches to better fit each of them. The Sequential Cascading Method processes patterns in parts: if the first fragment does not match, the rest can be ignored. The Vertical Join Method couples together different approaches or techniques in a single unit to provide higher efficiency of the resulting device. The optimization procedure maximizes efficiency gains for each method.*

The methods and methodologies presented in this study will allow developers to create more efficient reconfigurable tools for information security systems.

Keywords: *information security, FPGA, multi-pattern string matching, combination of approaches, efficiency, optimization.*

Отримано *5 серпня 2019 року*, затверджено редколегією *20 серпня 2019 року*